UNIVERSITA' DI GENOVA

SCUOLA POLITECNICA

Dipartimento di Ingegneria Meccanica

PhD Thesis

DECEMBER 2021

AA 2020 - 21

Immersive and Distributed Technologies for Simulation and Industrial Innovation

Relatore: Chiar.mo Prof. Ing. Agostino Bruzzone

Allievo: Marco Gotelli

Index

CHAPTER 2

Ringraziamenti

I miei ringraziamenti vanno a tutte le persone che mi hanno accompagnato e sostenuto durante la carriera universitaria e professionale durante gli ultimi anni.

In primo luogo, vorrei ringraziare la mia famiglia per il supporto incondizionato che mi ha fornito in questi anni. Durante le varie fasi di questo percorso mi hanno sempre sostenuto e motivato, indicando il modo migliore per raggiungere i miei obiettivi e costruirmi un futuro.

Ringrazio il mio relatore, il Professor Ing. Agostino Bruzzone, per l'interesse e la conoscenza della materia che ha saputo trasmettermi, per la fiducia che ha dimostrato e per avermi dato la possibilità di prendere parte ad iniziative ed esperienze che porterò sempre con me.

Un ringraziamento speciale va al mio amico e compagno di studi, Ing. Giuliano Fabbrini, che ha reso ogni sfida più semplice celebrando insieme ogni momento di soddisfazione.

Voglio infine ringraziare di cuore la Dottoressa Marina Massei e l'Ingegner Kirill Sinelshchikov per essere stati un punto di riferimento costante durante gli ultimi anni.

ABSTRACT

Today innovative technologies allow to develop new performing systems; therefore, also complex in terms of interactions among components and emerging behaviors. Usually, in these systems it is necessary not only to properly design and engineer the single component, but even the whole system, as well as the interactions with humans devoted to operate or to supervise with it; from this point of view digital technologies and simulation models are crucial and they allow to identify new and high performing configurations as well as requirements for man machine interface, innovative strategies and training programs. Indeed, in many cases the humans represent a critical element of the system itself as it is evident when we address the crew of a vessel or of a plane. Consequently, the human factors and digital technologies that enable people to develop new use cases and improve decision making are an essential component to drive innovation and design new systems; in this Thesis different cases have been investigated by using innovative simulation models and studying the application of digital technologies.

The first case is about an innovative immersive simulation system for training the crew of NH90 aircraft; the simulator is based on virtual reality technologies (e.g. head mounted display) integrated with an electromechanical motion platform, this layout creates an interactive and immersive training equipment, able to guarantee high quality training as well as compact and transportable quite low-cost solution. Furthermore, it has been developed an automated system to collect data about the user's response time to visual stimuli and it has been analyzed the possible negative training risk related to adopt these technologies in terms of impacts on human factors as well as on the simulation sickness phenomenon.

Furthermore, the application of Internet of Things and Cloud technologies will be analyzed to highlight relevant aspects of real use cases in the industrial sector.

Indeed, a functional study of the application of IoT to optimize resources in the industrial sector is presented with the proposition of a cloud architecture to automate the recognition of technical documentation.

INTRODUCTION

The data generated every day are approximately 2.5 Exabytes, and the Internet of Things and information technologies will most likely increase this number in the next years. Data come from heterogeneous sources like engineering and design processes, manufacturing, smart machines, supply chain, quality, sales, etc. The opportunities introduced by deducting meaningful information from digital sources are significant for any sector. Indeed, digital innovation outlines new paths for business models, operations improvement, and performance optimization.

In the last decades, digital technologies related to IoT (internet of things), advanced analytics, machine learning/AI, and immersive technologies (VR/AR/XR) are evolving very fast. The adoption of these technologies is also increasing at a high rate. Therefore, companies are pushed to invest in digital innovation to stay competitive, improve efficiency and evolve accordingly with the ever-changing conditions of the environment they operate in.

The solutions developed with these technologies, introduce tools that help take care of complex issues related both to technical and managerial aspects. Indeed, during re-engineering projects and technical innovation initiatives it is a common issue for teams to generate a reliable dataset to apply advanced analysis techniques or, when operating on a brown field to elaborate complex and unstructured data, both in historical databases and in real-time applications. These solutions directly impact all the functions of the organization, since they are aimed to improve information traceability, decision-making processes and help companies respect regulations supporting certifications with objective evidence. Furthermore, also the training processes are impacted by immersive technologies, whose capabilities will be analyzed in the first chapter together with a practical use case.

Accordingly, organizations and researchers are engaging in developing use cases that have a tangible effect on real-world problems, addressing critical aspects for staying competitive and enabling new growth opportunities, enhancing people's lives, and processes optimization.

Furthermore, digital applications enable the identification of relationships between different objects produced during operations. Drawings, models, design models, product evolutions often represent an important part of companies' knowledge, and they are part of a holistic environment that affects different departments of any productive company.

Complex products require the synergy of several systems to enable production. It is, therefore, crucial to organize and analyze the process and identify the parameters that affect the efficiency of the operations. To do so an important effort is needed from both engineers, subject matter experts, and technology developers.

Indeed, complex systems shaped on digital technologies introduce interaction on different levels and have an important impact on people designated to use them.

Moreover, it is very important to consider that most of these systems are based on humans. Indeed, research about human factors is fundamental as it aims to study the effect of technologies on people and their optimal design to be usable and efficient.

For this reason, M&S (Modelling & Simulation) is crucial to support the implementation and the adoption of new tools.

The human factor is part of every complex system whether innovating in industry or defense. When asking both the captain of a new vessel or an executive of a new business unit, which is the main component for success, the answer is often the people.

This thesis will investigate relevant aspects of real use cases applied to innovate through digital solutions in the industrial and defense sector. Indeed, this work aims to give a deep perspective of innovation results applying cutting-edge solutions and conducting a functional analysis to different issues that characterize complex problems and processes. Therefore, it is interesting to study and analyze different research streams (Immersive Technologies, IoT, and Cloud) to provide practical interrelation between research and innovation projects that bring a contribution to applied technologies.

In particular, the use cases will cover a study about VR training flight simulators and simulation sickness phenomena, IoT and the application of this technology for processes and resources optimization, and the utilization of a cloud-based application for digital documentation classification.

This thesis will present a detailed description of the tools that can be used to study such systems.

The software used for the VR simulator is VBS3 (Virtual Battle Space), software commonly used in the defense sector. The simulator is designed to reproduce an NH90 helicopter, the main features (technology, motion platform, etc.) are presented in Chapter 1.0. This part of the research aims to analyze immersive technologies application to train the crew of the helicopter and study the human factors related to the simulation sickness phenomena. In order to do so the training scenarios have been developed in VBS3 where, also, script devoted to the automatic logging of users' data has been developed (Chapter 2.6, 2.7).

IoT and cloud technology will be presented in chapter 3 where it has been carried out the study of the key feature of these technologies and the analysis of them. In Chapter 4 two uses cases are presented. The first one relates to the analysis of deploying energy sensors to enable IoT services devoted to monitoring and optimizing energy carrier's consumption. Therefore, it is a common factor for companies to develop reduction strategies on energy carriers to reduce the costs and respect environmental regulations. The integration of IoT, in this case, is particularly interesting since it is scalable, allows a high level of granularity, and enables reliable monitoring and predictive strategies. The cloud architecture presented in the second use case is based on Google Cloud Platform modules and it is aimed at the deployment of machine learning algorithms to automatically label and organize technical data (documentation, drawing, etc.). Indeed, during complex design procedures, it is a common issue to structure the information in order to realize processes smoothly and provide the correct documentation for users to be efficient during operations.

These studies have been carried out applying an analytical approach to the environment are applied in, describing the main characteristics of the technologies and the design process that is used to conceptualize such solutions.

CHAPTER 1

Extended Reality

The first activity carried out focused on an innovative simulator adopting immersive solutions and specifically in creating scenarios and measuring its effectiveness and capability respect training considering the human factors

The application of this technology will be applied to a flight simulator. Aerospace is a sector where the simulation is widely used since over half a century: traditional applications are related to engineering and training. In facts, simulation is used for training purposes in aeronautics and aerospace to support the development of advanced skills and techniques respect complex missions. This aspect is common in civil area, specifically to learn psychomotor and cognitive aspects related to flying and operating a plane, or helicopter as well as a drone, but it turns even more crucial in references to missions dealing with defense, homeland security and civil protection where the operations are much more sophisticate than just flying, requiring complex tasks often in strict collaboration with other entities.

Due to these reasons the flight simulators are extensively used to train pilots across all branches of the Services and Units (e.g. firefighters, civil protection). They are used to teach flying skills, operational missions as well as how to manage emergency and to communicate with ground control.

However modern technologies are generating a fast evolution in training equipment in this field, for instance in addition to very expensive traditional flight simulators many new lean solutions are appearing: head mounted displays, augmented reality glasses, compact motion platforms and other technologies allow today to develop new small flight simulator potentially pretty effective. Therefore, these innovative solutions could introduce criticalities respect human perceptions and, consequently, training effectiveness. Indeed, this part of the Thesis will focus on the evaluation of these human factors respect training helicopter crew in missions; the activity involved the creation of models and scenarios able to outline these aspects with specific attention to the case of a multi-role helicopter in use by German Air Force respect R&D Activities carried out at the FKIE in Bonn.

This new simulator takes advantage of the innovative technologies of the virtual reality and related immersive technologies. Indeed, by creating a virtual environment it is possible to recreate real situations and get a more immersive and effective training with low costs respect traditional full scope simulation. Moreover, the simulator has an innovative layout: it is uncommon to find a helicopter simulator such as that one experimented, that combines virtual reality with real hardware controls and a motion platform; this approach guarantee also to create a solution pretty compact and easy to redeploy in a much easier way respect traditional simulator.

Furthermore, it has been investigated the response of a trainee exposed to VR simulation and its own perceptions through tests and experiments; indeed, the proposed analysis includes also the concept of simulation sickness and a methodology devoted to evaluating it.

Since the helicopter is an interconnected system, the concept of SoSE (System of System Engineering) assumes an important aspect; indeed, in this simulator it will be possible to interact between the crew and the ground troops as well as the between the crew and the interaction between the pilot and the implemented system of the helicopter that will be presented below.

1.0 State of Art

Improving processes with immersive technologies is an opportunity that companies, characterized by complex production processes and product development, are taking advantage of (Berg & Vance, 2017). Nowadays, software and hardware developments have been greatly fast, allowing high technology to be deployed in the market for significantly beneficial use cases. Nowadays, Immersive Technologies, such as Extended Reality (XR), Mixed Reality (MR), Virtual Reality (VR), Augmented Reality AR), are creating new opportunities. Indeed Immersive Technologies, also known as Digital Reality, jointly with the Internet of Things (IoT) and complex control systems are paving the way for huge innovations. Digital Reality allows users to interact with a virtual environment in an apparently physical way using wearable (i.e. HMD) or mobile devices. Currently, there are some significant use cases in the industrial sector regarding several topics: remote collaboration, immersive and collaborative training, data visualization, and design. The benefits affect the whole process starting from the enhancement of the product design to the improvement of proficiency, supply chain planning, and workforce collaboration. These have had a real boost in the last years solving some of the inherent problems of these technologies. Headsets are especially experiencing great innovations. VR devices are decreasing the image latency at a high rate, the enhancement of mobile computing will soon eliminate the need to connect, and new breakthroughs in tracking objects, body parts, and movements are opening the way for a whole set of applications.

1.0.1 Overview

The technologies used to create immersive experiences are becoming more and more sophisticated and realistic, while at the same time their cost are downgraded making it possible to extend their use in many more application areas (Hale & Stanney, 2014; Bruzzone et al.,2016a; Berg & Vance, 2017). Along last years we moved from the original 1 million USD cost of a CAVE (Cave Automatic Virtual Environment) around 20 years ago to 40,000 USD for the basic configuration of a modern compact interoperable and touchscreen system such as SPIDER (Simulation Practical Immersive Dynamic Environment for Reengineering) introduced by Simulation Team (Hereld et al., 2000; Cruz-Neira et al.,2013; Bruzzone et al.,2016b). What is interesting concerning SPIDER is that even if visualization just full HD over 2m wide screens surrounding the people inside the cube, it introduces two very innovative concepts at a low price:

- touch screen solution that enables to touch the screens all around SPIDER users (in the above mentioned basic configuration over 270□ horizontally within the cubicle of 2m x 2m x 2.6m) and interact with objects
- fully interoperability allowing to combine the SPIDER with multiple models and simulators as part of a Federation

This means that even low-cost solutions nowadays introduce very high-value new features and capabilities compared to old classical expensive solutions that were limiting the use mostly just to defense and aerospace (Vandervliet 1992; Oberhauser et al.,2015)

In addition, the developments in the field of VR & AR devices have created a scalable set of solutions that moves from 400,000 USD of an advanced Visualization System to 40,000 of a SPIDER, 4,000 for AR with Hololens, 400 for VR with Oculus Rift down to less than 40 Euro for a Headset holder for a Smartphone able to support stereoscopic viewing and VR (Papachristos et al.,2017; Bruzzone et al. 2019; Elor et al.2020).

Obviously, in case of distributed interactive simulation, the adoption of correct protocols and exchanges of data, as well as techniques to improve connectivity performance, is essential (Bruzzone et al., 1998, Zeigler et al., 1998; Fujimoto 2000; 2017; Liu et al., 2014; Graf et al., 2017).

So today, there are many mixes of solutions that, based on the different application and user, results to be the most effective.

1.0.2 Devices

Smartphones and tablets are the simplest device to be used due to their wide adoption. By the way, it is possible to create valuable experiences when they do not need a high level of complexity in operations and graphics (Steed & Julier, 2013; Ko et al.,2013).

These devices are mostly used for augmented reality. Augmented Reality (AR) overlays virtual and real world. For example, it could introduce text, images and animations on top of physical objects. Mixed Reality (MR) term is used to describe more interactive AR (Benford & Gianmachi, 2011).

Interesting use cases for AR are the ones where operators and experts are geographically spread (i.e. Remote Maintenance, remote commissioning, etc.). Today VR and AR could provide major benefits in remote service and maintenance as demonstrated by recent researches by the authors (Bruzzone et al.,2016d). Indeed, developing a web application based on mobile devices that tracks objects and overlap virtual ones can be a competitive advantage, allowing companies and agencies to reap a huge amount of money and improve communication in tough situations (Bruzzone et al.,2016a; Mourtzis et al.2017).

For Virtual Reality the set of equipment includes, but not limited to Head Mounted Displays (HMD), projected environments, controllers, motion platforms, motion tracking systems, haptic gloves and suits. Principal fields of application of VR are education, training and entertainment, but also support of engineering and safety preparation (Jacobson & Hwang 2002; Hale & Stanney, 2014); by the way recently even autonomous systems are turning in additional elements of modern XR for multiple purposes (Bruzzone et al., 2016c; Kim et al., 2018). However, technologies and objectives are evolving and changing rapidly (Muñoz-Saavedra et al.,2020).

VR solutions are used for training of operators of different kind of machinery in industry (e.g. cranes), facilities (e.g. industrial plants) as well as for recreational (e.g. museums & exhibitions) & educational purposes in classrooms (Whisker et al., 2003; Bruzzone & Longo, 2013; Mastli & Zhang, 2017; Longo et al.,2018; Liu et al.,2020; Puig et al., 2020).

Most products include not only headsets, but also controllers, capable to provide interaction capabilities with the virtual world (Sagayam & Hemanth, 2017; Kumari & Polke, 2018). For example, many controllers allow tracking not only position but also spatial orientation, which can be used to ‘point’ on virtual objects, drag and drop them. Such systems can include different set of inputs, starting from one simple click button and up to combination of joysticks, buttons and touch sensors capable to detect “near click” (Bowman et al., 2012; Coelho et al., 2014).

There are solutions that are also non wearable. The most common are called CAVES (Cruz Neira et al., 2012). CAVES have been used in many different fields, from military training to medicine to visualize parts of a body giving so an opportunity to prepare for operation in shared environment (Hale et al. 2014). In addition to these fields there are CAVE used in Universities and Industries as virtual show rooms, or in Museum for the reproduction of natural or past environment (Muhanna, 2015). In facts, some applications of CAVE are specifically related to entertainment sector (Jacobson & Hwang 2002). In facts, a CAVE contains usually a limited space where virtual world is reproduced, but it allow the users to enter and eventually, by most modern solutions, to interact with it (Hale et al. 2014; Bruzzone et al. 2016a). Images could be created using classic direct view otherwise rear projection, which reduce drastically number of components inside CAVE, hence improving its virtual immersion, however this solution is not so comfortable in exploitation and requires bigger external volume and space occupancy of the whole equipment (Hale et al. 2014).

1.0.3 Motion Platform

The user experience of both the HMD and Caves can be enhanced through the integration of a motion platform (Advani & Va, 1995; Chen 2001; Thöndel 2010; Lee 2020). They allow simulation of vibration, collisions and accelerations. It is possible to employ in virtual reality even more specific or specialized equipment (VanderVliet 1992; Pollini et al.,2008). In practice, nowadays more and more devices are being developed with VR integration in mind. In the following lines is shown a specific motion platform that has been used for helicopter simulation that represent an interesting application field (Schroeder, 1999).

An example of a 6 DOF (Degrees of Freedom) motion platform based on Steward Platform (in this case courtesy of Brunner for model Motion1000 – see Figure 1.28); the corresponding motion control system provides different motion cueing algorithms, allowing adapt the device for different types of simulations. A motion platform represent an interesting opportunity for improving fidelity of the flight simulator and it reduces discrepancies between the vestibular and the visual system during the virtual reality simulation; therefore in flight-simulators this is on the most sickening factors. It is important to outline that Artificial Intelligence has a great potential in reducing the impact of motion sickness by developing innovative control systems able to mitigate it though prediction of users movements (Hell et al., 2018). Therefore it is always necessary to remember that is fundamental to properly evaluate the advantage provided by motion platform respect the specific objective of the simulator (Amico et al.,2000). Further details on the motion platform used to develop a helicopter simulator are presented in Chapter 1.2.4.

1.0.4 Tracking systems

The applications that imply a physical movement of one or more users need a tracking system (Chang et al., 2001; Cameron et al.,2011; Shin et al.,2016; Park et al., 2017). As aforementioned, HMDs have their own tracking system for a narrow area. For custom applications or for expanding the available walking area there are different technologies for navigation and positioning.

In the following lines, different type of devices will be analyzed:

- Inertial measurements units (IMU): They are devices capable to detect angular and linear accelerations (Zhang et al., 2015). These sensors are usually relatively cheap and allow to provide minimum positioning to various devices, such as smartphones. These systems do not depend on external services or infrastructures. IMUs are particularly useful for dead reckoning – to guarantee positioning even after external connections are lost (e.g. GPS after enter in tunnel). The principal component of any IMU are accelerometers and gyroscopes. It possible to find different types of accelerometers and gyroscopes , for instances in mobile systems they are very small, yet they provide high level of precision
- GNSS (Global Navigation Satellite System): One of most known systems for outdoor positioning is GPS, which is used in most of modern mobile phones. It allows to identify coordinates in most of the world, even if final precision could be not sufficient for certain purposes. There are several projects which employ positioning data obtained from GNSS and gyroscopes in augmented reality applications, for instance, to extend functionality of navigators (Angelino et al.,2012; Wang et al.,2012).
- Radio beacons: In some situations it is convenient to utilize general purpose indoor positioning systems (Foxlin et al.,1998); for instance, they often offer

higher range of operations respect custom solutions. Currently, several different technologies are used for indoor positioning, most diffused are:

- Bluetooth
- UWB (ultra-wide band)
- Wifi
- Constellation tracking: Each device is equipped with unique configuration of IR LEDs, visible only to tracking stations, which are capable to individuate position and orientation of each piece of equipment. This type of tracking requires brief installation and configuration of equipment following interactive instruction. Positioning is based on external sensors so proper positioning is require.
- Lighthouse: This method, deploys beacons to produce timed IR pulses and X/Y axis IR laser sweeps. Lighthouse integrates IR-filtered photodiodes inside anything that need to be tracked. Opposed to constellation tracking, it does not use IR LEDs affixed on the HMD and controllers. The signals received from the diodes are amplified and sent into an integrated circuit.
  The circuit is programmed with the relative location where each input signal was obtained, as the X and Y 'plane' of IR laser light sweeps across the numerous sensors contained inside the controllers and HMD.
- Inside-out: This system is used by Microsoft to map and fix holograms position using Hololens. The system is based on two black and white cameras that identify the characteristics of the real environment. The data are then fused continuously with high rate IMU data to update the position of the device in the environment.
- Focusing on Augmented reality, modern toolkits allow using various objects as positing tags: QR codes, images, surfaces with patterns (Pence, 2010). Tag recognition starts by detecting “features” like sharp or spiked details image.

It is highly recommended that such tags have high contrast, elevated number of features and avoid too repetitive patterns (e.g. grids). The disadvantage of this approach is that false positive and false negative detection has relatively high probability and that image targets must satisfy basic requirements.

In virtual reality headset, the tracking system continuously maps the environment around and tries to correlate it with known ones, consequently understanding its relative position.

From the experience in industrial projects in emerged a difficulty in operation in following conditions:

- Presence of multiple moving obstacles which obstruct positioning, such as human operators or conveyor belts
- Operation in zones with wide empty areas, for example free part of a warehouse and very big rooms, which overwhelm capacity of recognition of 3D vision and depth perception sensors

Tracking users' movements enhance these applications also for motion capture. Motion capture data in immersive experiences might be used both to register real user behaviors and reproduce them in avatars and to build predictive models. Indeed, predictive algorithms applied to tracking systems can improve the system performance forecasting in real-time which will be the users movements during the experience. Some examples of motion capture are:

- Mechanical or magnetic: Physical sensors are attached to the body.
- Optical-passive: Is based on reflective markers tracked by cameras, usually operating in IR. High precision, widely used.
- Optical-active: Similar to the previous one but uses LED markers.
- Video marker less: Software-based tracking of objects in video feed.

- Inertial: Position of objects is dead reckoned by IMU

It is particularly challenging to track in high fidelity the hands movements. Controllers provide a basic tracking. For example the HMDs controllers can detect whether a hand is open or not and basic movements of fingers. A technological solution that is being widely is the Leap Motion is an optical hand-tracking device that captures the movement hands. The Leap Motion is mounted on the front part of HMDs; this causes the main shortfall of the device. Indeed, the leap motion tracks the movement only when the user is looking at their hands. By the way, it constitutes a cost effective and precise solution.

1.0.5 Free Roaming

Free-roaming allows trainees/users to move freely within a room by tracking their movements (Wang et al.,2011). In fact, the system reproduces exactly the movements in a parallel virtual environment that remains aligned to the action of the people involved and any virtual twin of physical objects, whether real or mockup; this allows users to interact with each other and with the simulation. These systems can work in an integrated way with Virtual reality using free roaming as an added value to experiment different scenarios for training or decision support. It is possible to train trainees to respond to sudden events or perform maintenance operations of complex equipment or damaged vehicles in a virtual environment.

The possibility of interaction between one trainees/user and the other is a breakthrough, combining Virtual Reality (Free Roaming compared to common Virtual Reality systems).

Below are listed the main advantages of this approach:

- Tracking of body movements including upper and lower limbs and fingers. (Ex: The trainee has the ability to follow the movements of his hands within the virtual environment)

- Social Interaction of Trainees: Trainees can cooperate thanks to the creation of avatars to which real movements are associated, tracked by trackers.
- Interaction with virtual objects: the same trackers can also be placed on physical devices, associating them to virtual objects that are reported within the scenario
- Modularity: the system can manage from 1 to 6 trainees
- Space Scalability: You can develop flexible applications operating on different scenarios and sizes. In the following the technical description is based on a reference size of 25 m2, but of course it is possible to extend and scale the system for larger spaces with additional devices and upgrades.
- Retention Rate: Through the use of virtual reality with free-roaming is guaranteed a faster learning compared to traditional technologies. In how much it is tied up the training to particularly immersive experiences, where the proprioceptive elements and of psicophysical fellowship are emphasized from the tracking of the behavior of the subjects and their body during the simulated activities.
- Overcoming spatial constraints: The key point of the innovativeness of the system lies in the fact that the user, or trainee, will not have the constraints of movement imposed by traditional VR systems, as the free-roaming system is Wireless and, through tracking, allows you to walk, move, climb on objects, move between different rooms allowing, possibly, the extension on articulated spaces (e.g. inside a building, plant or vessel).
- Transportable: the system can be easily disassembled and assembled to be moved according to needs.
- Detection of biometric parameters (e.g. muscle tone, heartbeat) through an optional module that can be integrated.

The movements need to be tracked and the data have to be sent to the local computer. In order to do so, it is proposed the following scheme:

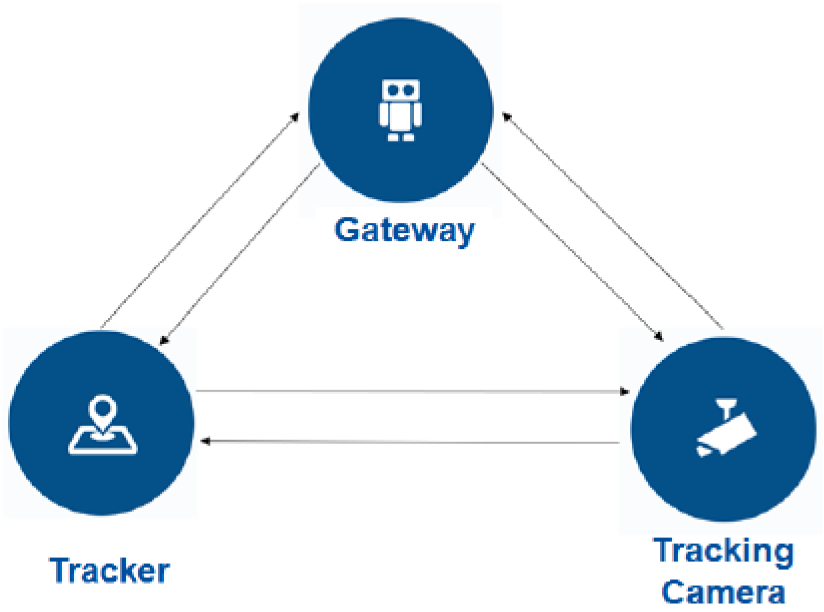


Figure 1.1 Components Scheme

The body trackers will be positioned in the following way:

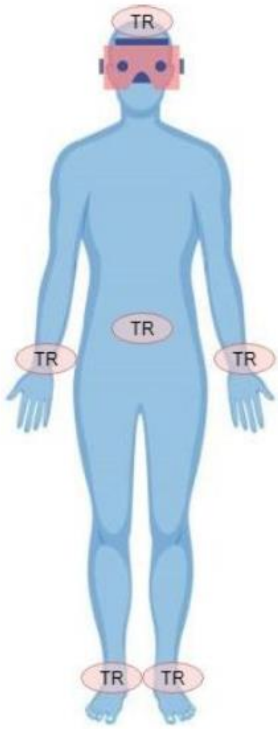


Figure 1.2 – Body sensors

The tracking camera follow the tracker mounted on the users like in Figure 3. An example of the tracking cameras for a default space of 25 square meters is shown in the figure below:

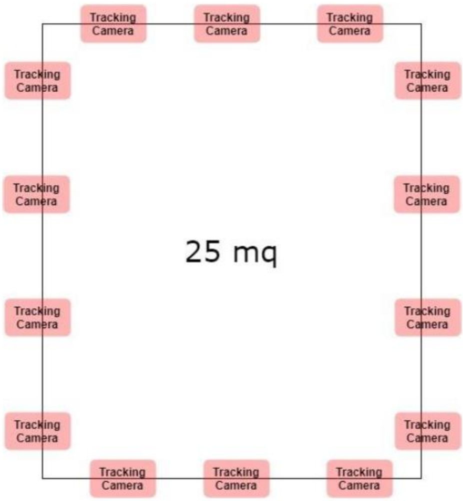


Figure 1.3 – Tracking Cameras

1.1 Flight Simulator

The helicopter model was developed by the Bundeswehr. In the image below is reported a figure of the model in VBS3.

Figure 1.4 – NH9 model

The most important features of the flight simulator are:

- Multiplayer Simulator
- Virtual Reality Simulation with HMD
- Real Hardware for flights controls

- Motion Platform

1.1.1 Multiplayer Simulator

The simulator has been designed for training two people at the same time. One will be placed at the pilot position, the other will stay at the crew position.
They will be able to communicate each other and the performance of one will influence the one of the other.

1.1.2 Virtual Reality Simulation with HMD

This simulation has been made with virtual reality technology.
Two different Head-mounted display (HMD) have been used:

- Oculus Rift
- HTC Vie

There are two HMD because in this work there is also a multiplayer simulation.
Before to speak about HMD, a brief introduction of Virtual Reality is presented.

<u>Virtual Reality</u>

Virtual Reality is a computer technology that uses virtual handsets or multi-projected environments, sometimes in combination with physical environments or props, to generate realistic images, sounds and other sensations that simulate a user's physical presence in a virtual or imaginary environment. A person using virtual reality equipment is able to explore the artificial world, and with high-quality VR move around in it and interact with virtual features or items. The immersive environment can be similar to the real world in order to create a lifelike experience grounded in reality or sci-fi.

In other words, it's possible to define the Virtual Reality in this way:

Inducing targeted behavior in an organism by using artificial sensory stimulation, while the organism has little or no awareness of the interference. (Steven M. La Valle, 2017)

Four key components appear in the definition:

- Targeted behavior: The organism is having an "experience" that was designed by the creator. Examples include flying, walking, exploring, watching a movie, and socializing with other organisms.
- Organism: This could be a person or even another life form such as a fruit fly, cockroach, fish, rodent, or monkey.
- Artificial sensory stimulation: Through the power of engineering, one or more senses of the organism become hijacked, and their ordinary inputs are replaced by artificial stimulation.
- Awareness: While having the experience, the organism seems unaware of the interference, thereby being "fooled" into feeling present in a virtual world. This unawareness leads to a sense of presence in an altered or another world. It is accepted as being natural.

In order to work with virtual reality there are two important things to know:

- Hardware
- Software

Hardware

The first step to understanding how VR works is to consider what constitutes the entire VR system. It is tempting to think of it as being merely the hardware components, such as computers, headsets and controllers. This would be woefully incomplete. As shown in Fig 1.5, it is equally important to account for the organism, which in this chapter will exclusively refer to a human user. The hardware produces stimuli that override the senses of the user by using its own sensors, thereby tracking motions of the user. Head tracking is the most important, but tracking also may include button presses, controller movements, eye movements, or the movements of any other body parts. Finally, it is also important to consider the surrounding physical world as part of the VR system. Despite stimulation provided by the VR hardware, the user will always have other senses that respond to stimuli from the real world. The VR hardware might also track objects other than the user, especially if interaction with them is part of the VR experience. Through a robotic interface, the VR hardware might also change the real world.

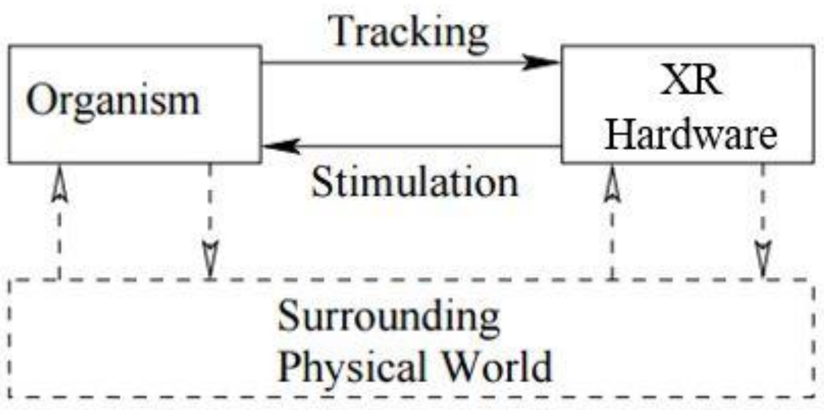


Figure 1. 5 (Steven M. La Valle, 2017)

From a developer's standpoint, it would be ideal to program the VR system by providing high-level descriptions and having the software automatically determine all of the low-level details. In a perfect world, there would be a VR engine, which serves a purpose similar to the game engines available today for creating video games. If the developer follows patterns that many before her have implemented already, then many complicated details can be avoided by simply calling functions from a well-designed software library. Unfortunately, we are currently a long way from having fully functional, general-purpose VR engines.

In order to produce a Virtual Reality experience many components are necessary, the Fig 1.6 presents a high-level view that highlights the central role of the Virtual World Generator (VWG). The VWG receives inputs from low-level systems that indicate what the user is doing in the real world. A head tracker provides timely estimates of the user's head position and orientation. Keyboard, mouse, and game controller events arrive in a queue that is ready to be processed. The key role of the VWG is to maintain enough of an internal "reality" so that renderers can extract the information they need to calculate outputs for their displays.

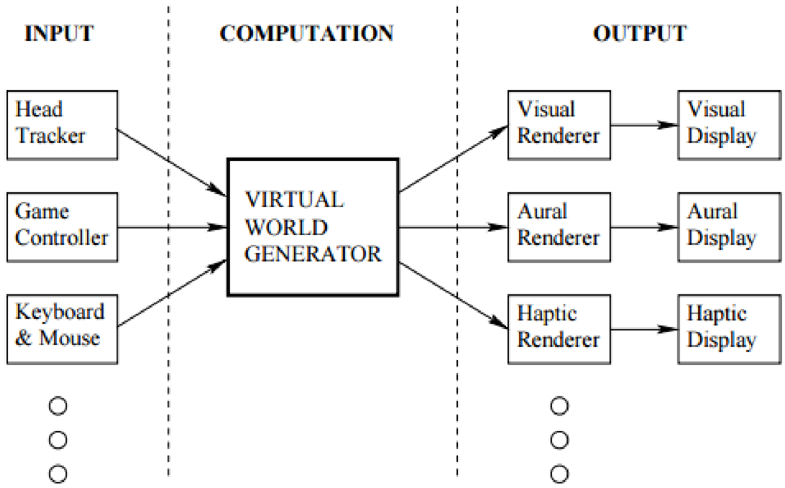


Figure 1 .6 (Steven M. La Valle, 2017)

It is only a brief explanation of the Virtual Reality but it is useful for understanding which kind of technology has been used in this work.

Head-mounted display (HMD)

This is a display device, worn on the head or as part of a helmet, that has a small display optic in front of each eye HMD has many uses including in gaming, aviation, engineering, and medicine.

For the simulations, an optical head-mounted display has been used. It can reflect artificial images, and let real images cross the lens, and let a user look through it.

<u>Oculus</u>

It is developed and manufactured by Oculus VR. The Rift has a Pentile OLED display (matrix schemes used in electronic device displays, 1080×1200 resolution per eye, a 90 [HZ] refresh rate and 110° field of view. It has implemented headphones which provide a 3D audio effect (this frequently involves the virtual placement of sound sources anywhere in three-dimensional space, including behind, above or below the listener). (www.oculus.com)

This is the main difference between the HTC Vive. The Oculus Rift is available for the trainer that will be in the crew position.

Also, the Oculus Rift has integrated rotational and positional tracking. The positional tracking system, called "Constellation", is performed by a USB stationary infrared sensor that is picking up light that is emitted by IR LED that is integrated into the head-mounted display.

The sensor has to sit on the user’s desk. This creates 3D space, allowing for the user to use the Rift while sitting, standing, or walking around the same room.

Figure 1.7 (www.oculus.com)

In the Fig 1.7 is shown how the Oculus Rift looks like.
In order to use the Oculus Rift the castellation have to be installed (Fig 1.5), it is the headset's positional tracking system used to track the position of the user's head as well as other VR devices, consisting of external infrared tracking sensors which optically track specially designed VR devices. The constellation sensor comes with a stand of a desk lamp form factor, but has standard screw holes and can be detached from this stand and mounted anywhere appropriate to the user.
In the laboratory two castellations have been installed, Oculus will allow third-party peripheral manufacturers to create their own devices that are tracked by the system, providing an API for them to use.

Figure 1. 8 (www.oculus.com)

There are also two controllers, they were not used in the simulations. This motion controller system consists of a pair of handheld units, one for each hand, each containing an analog stick, three buttons, and two triggers. The controllers are fully tracked in 3D space by the Constellation system, so they may be represented in the virtual environment

The controller looks like in the Fig 1.9

Figure 1.9 (www.oculus.com)

HTC

This device is similar to Oculus Rift and it works in the same way.
The main differences are:

- Two cameras have to be installed in the corner of the room, instead of the castellation.
  The Lighthouse tracking system is composed of two black boxes that create a 360° virtual space up to 15x15 foot radius. The base stations emit timed infrared pulses at 60 pulses per second that are then picked up by the headset and controllers with sub-millimeter precision.

- Vive HTC has not implemented headphones

- Different kinds of controllers: The controller has multiple input methods included a trackpad, grip buttons, and a dual-stage trigger and a use per charge of about 6 hours. Across the ring of the controller, there are 24 infrared sensors that give the base station the location of the controller. The Steam VR Tracking system is used to increase the connection of the controller by giving wireless real-time feedback of 360° to the host.

- There is a special tracker that brings real-life objects and takes them into the virtual world. With a six pin connector that can connect the tracker to many objects. The Vive tracker functions like the Vive Controllers and the Vive Headset by gathering information from the infrared emitted by base stations. Items on the tracker include LED indicator, power button, friction pad and micro USB. In this simulation, two cameras are in the up corners of the room.

- The headset has a camera. The software can use the camera to identify any moving or static objects in a room; this functionality can be used as part of a "Chaperone" safety system which will automatically display a virtual wall or a feed from the camera to safely guide users from obstacles or real-world walls.

- The resolution is 2160x1200 combined pixels.

Figure 1.10 (www.vive.com)

In the Fig 1.10 is shown how the HTC Vive looks like.

1.1.3 Helicopter controls

The helicopter fly controls are not included on the moving platform.
In order to complete the work these controls have to buy; the Flight Link LLC company has been chosen.
The Flight Link LLC is located in the US and it produces flight simulators, controls and aviation training.
In this work, the following controls have been bought:

- Cyclic control
- Collective pitch control
- Anti-torque pedals

The control panel with all indicator is present in the VBS simulation. It was not necessary to buy it.
Before to explain each control, it is useful to understand how to fly the helicopter.
In the next lines, there is a brief explanation.
A helicopter pilot manipulates the helicopter flight controls to achieve and maintain controlled aerodynamic flight. Changes to the aircraft flight control system transmit mechanically to the rotor, producing aerodynamic effects on the rotor blades that make the helicopter move in a deliberate way. To tilt forward and back (pitch) or sideways (roll) requires that the controls alter the angle of attack of the main rotor blades cyclically during rotation, creating differing amounts of lift (force) at different points in the cycle. To increase or decrease overall lift requires that the controls alter the angle of attack for all blades collectively by equal amounts at the same time, resulting in ascent, descent, acceleration and deceleration.
A typical helicopter has three flight control inputs the cyclic stick, the collective lever and the anti-torque pedals. Depending on the complexity of the helicopter, the cyclic and collective may be linked together by a mixing unit, a mechanical or hydraulic

device that combines the inputs from both and then sends along the "mixed" input to the control surfaces to achieve the desired result. The manual throttle may also be considered a flight control because it is needed to maintain rotor speed on smaller helicopters without governors. The governors also help the pilot control the collective pitch on the helicopter's main rotors, to keep a stable, more accurate flight.

In the next pages, there will be an explanation of each control, in order to understand how they work some helicopter components will be mentioned.

In Fig 1.11 the main components of a helicopter are shown

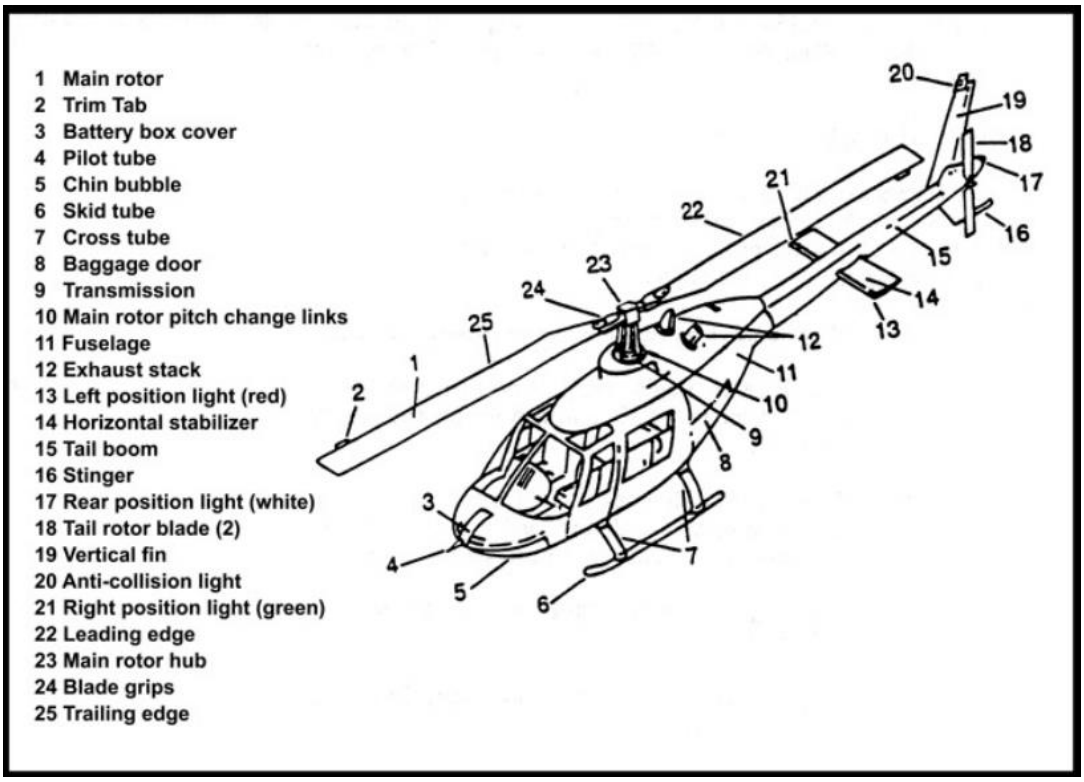


Figure 1. 11 (Minister of transport Canada, 1996)

Cyclic control

The cyclic control commonly called the cyclic stick or just cyclic, it is located between the pilot's legs.

The control is called cyclic because it changes the pitch angle of the rotor blades cyclically. That is, the pitch, or feathering angle, of the rotor blades changes depending upon their position as they rotate around the hub so that all blades have the same incidence at the same point in the cycle. The change in cyclic pitch has the effect of changing the angle of attack, and thus the lift generated by a single blade as it moves around the rotor disk. This, in turn, causes the blades to fly up or down in sequence, depending on the changes in lift affecting each individual blade.

The result is to tilt the rotor disk in a particular direction, resulting in the helicopter moving in that direction. If the pilot pushes the cyclic forward, the rotor disk tilts forward, and the rotor produces a thrust vector in the forward direction. If the pilot pushes the cyclic to the right, the rotor disk tilts to the right and produces thrust in that direction, causing the helicopter to move sideways in a hover or to roll into a right turn during forwarding flight, much as in a fixed wing aircraft.

A rotor is an oscillatory system that obeys the laws that govern vibration, depending on the rotor system, may resemble the behavior of a gyroscope. (Minister of transport Canada, 1996)

In the Fig 1.12 the influence of the cycling control is shown.

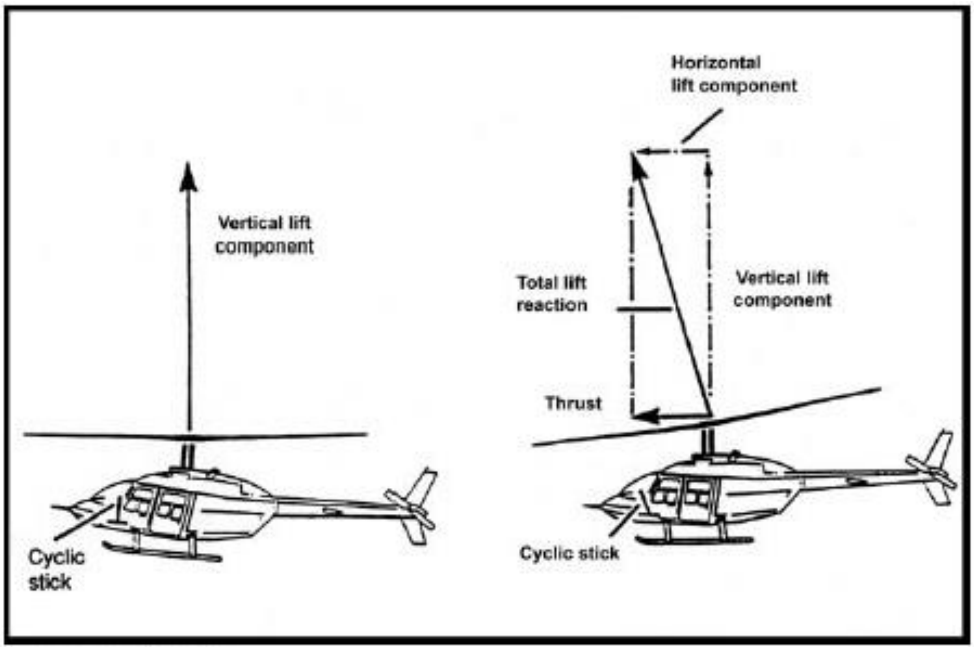


Figure 1. 12 (Minister of transport Canada, 1996)

The control that will be installed in the moving platform is shown in the Fig 1.13 and in the Fig1.14

Figure 1.13

Figure 1. 14 (www.flightlink.com)

In the Fig 1.15 and Fig 1.16 the real cyclic control is shown.

Figure 1.15

Figure 1.16

Comparing the Fig 1.13 and the Fig 1.16, it is possible to see that many buttons are in a different position.

It is a limitation of the simulation. Anyway, all buttons present in the Fig 1.13 are non-necessary for flying.

Collective pitch control

The collective pitch control, or collective level, is normally located on the left side of the pilot's seat with an adjustable friction control to prevent inadvertent movement. The collective changes the pitch angle of all the main rotor blades collectively (i.e., all at the same time) and independent of their position (Minister of transport Canada, 1996), like in the Fig 1.17

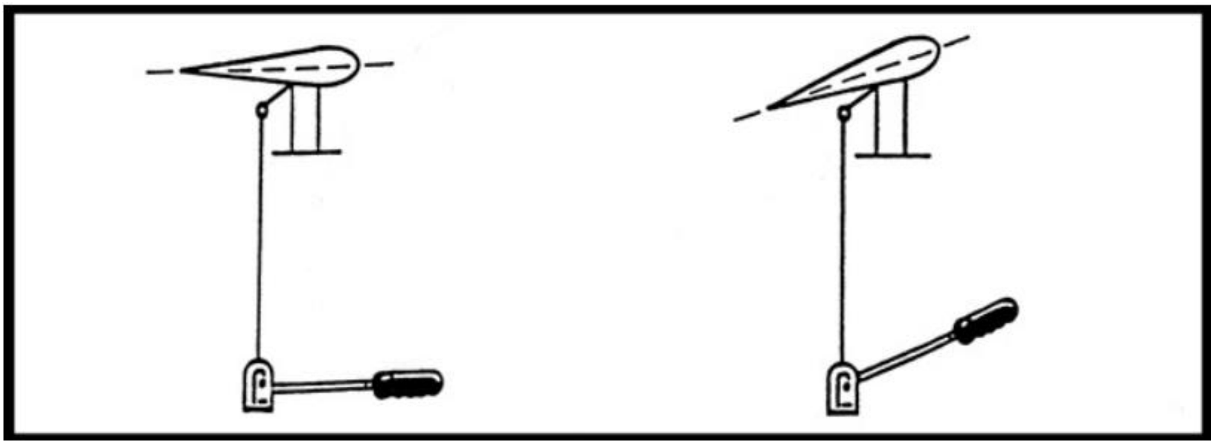


Figure 1 .17 (Minister of transport Canada, 1996)

Therefore, if a collective input is made, all the blades change equally, and as a result, the helicopter increases or decreases its total lift derived from the rotor. In level flight, this would cause a climb or descent, while with the helicopter pitched forward an increase in total lift would produce an acceleration together with a given amount of ascent.

In addition to this command, there is a throttle.

A twist grip throttle is mounted on the forward end of the collective. It is used to set the engine and rotor rpm to the normal operating. Applying throttle will also cause the manifold pressure to increase and the helicopter to yaw to the right. Reducing throttle, while causing a decrease in rpm and manifold pressure, will again cause yaw, this time to the left.

In the Fig 1.18 the control that will be installed on the moving platform is shown.

In the Fig 1.19 the real NH90 control is shown.

Figure 1.18(www.flightlink.com)

Figure 1.19

The difference between the real and non-real control is a lot in this case. Anyway, the trainee thanks to the HMD device will see the real collective pitch control.

<u>Pedals</u>

The purpose of the pedal controlled tail rotor is to counteract the torque effect of the main
rotor, to control the heading of the helicopter during hovering flight, and to initiate turns while
in the hover. It is not, however, used to control the heading while in cruise flight, but only to
compensate for torque. This puts the helicopter in longitudinal trim so as to maintain co-ordinated flight (Minister of transport Canada, 1996) (Fig 1.20).

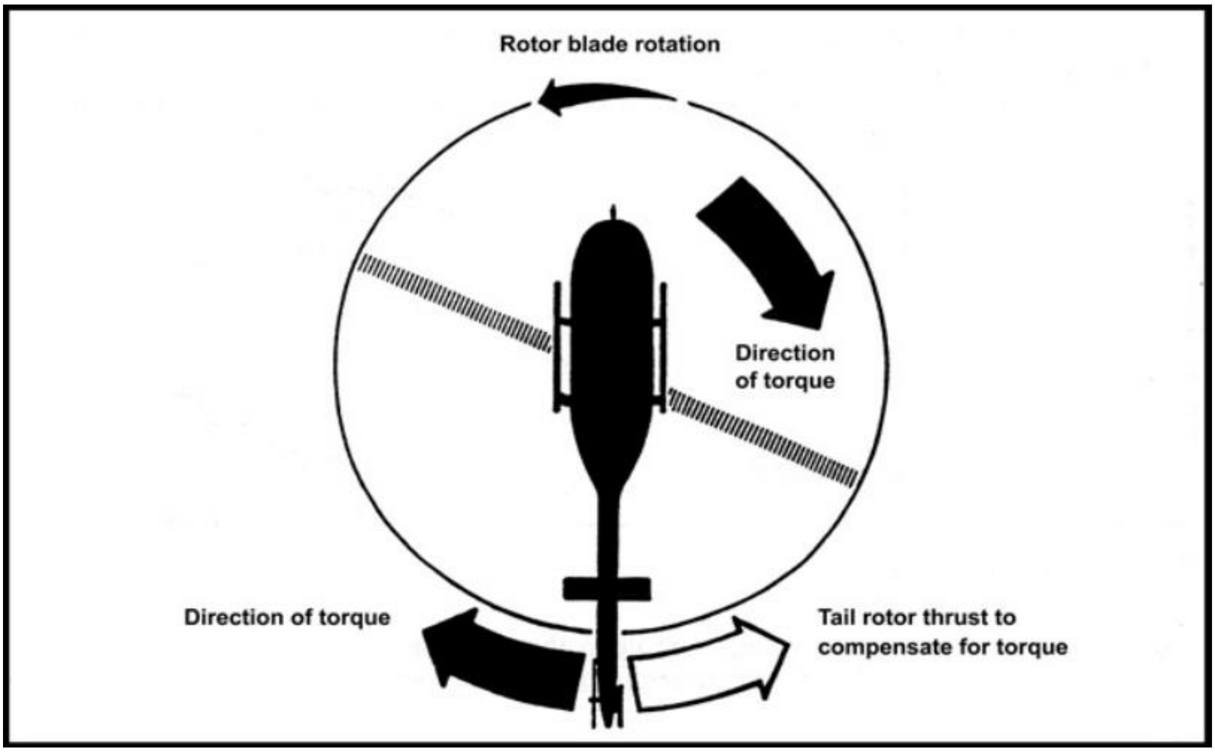


Figure 1.20 (Minister of transport Canada, 1996)

Movement of the pedals will effect a change in the collective pitch of the tail rotor blades. The result of pressure on one pedal will be a yaw in the corresponding direction, for example; pressure on the left pedal will cause the nose to yaw left, and vice versa. An increase in pitch will require an increase in power, a decrease in pitch a decrease in power.

Any change in collective or cyclic position will require adjusting the pedals in order to maintain co-ordinated flight.

In the Fig 1.21 the non-real pedal is shown.

Figure 1. 21 (www.flightlink.com)

The Flight Link LLC provides a seat and box for connecting the above-mentioned components.

In the Fig 1.22 all components are shown.

Figure 1.22 (www.flightlink.com)

All the components shown in Fig 1.22 will be installed in the monition platform.

Control panel

There are many instruments in the control panel of a real helicopter.
In the following lines, the main instruments are mentioned.

Fly Instruments: (Minister of transport Canada, 1996)

- Airspeed Indicator: This instrument indicates the speed of the helicopter through the air in which it's flying; it relates only indirectly to the speed of the helicopter over the ground. It may indicate speed in miles per hour or knots.

- Altimeter: This is a pressure sensitive instrument that, if properly set, indicates the
altitude at which the helicopter is flying.

- Turn And Bank Indicator: The needle portion of this instrument indicates whether the helicopter is turning, together with the direction and rate of turn. The ball portion of the instrument is fundamentally a reference for coordination of controls.

- Magnetic Compass: This is the basic reference for heading information. The compass correction card indicates the corrected heading to steer to allow for compass deviation.

- Attitude Indicator: It provides the pilot with an artificial horizon, which together with a miniature aircraft superimposed on its face enables the pilot to determine the aircraft's attitude relative to the real horizon.

- Vertical Speed Indicator: This is a pressure sensitive instrument, which indicates the rate at which the helicopter is climbing or descending in feet per minute.

Engine Instruments: (Minister of transport Canada, 1996)

- Dual Tachometer: This instrument indicates the number of revolutions per minute (RPM) that both the engine (ERPM) and the rotor blades (RRPM) are making.

- Manifold Pressure Gauge: This instrument is calibrated in inches of mercury and indicates the pressure in the intake manifold of the engine. Stated more simply, it indicates the amount of work the engine is doing. It is normally expressed as a percentage with 100% being the maximum available, but in some helicopters, foot/pounds are the measures used.

The control panel in the simulation is very similar to the real control panel.
In the Fig 1.23 and Fig 1.24 both are shown.

Figure 1.23 – Real Control Panel

Figure 1.24 – Simulated Control Panel

In the simulation, these instruments are used only to give the impression of being in a real helicopter. They do not indicate anything.
In the simulation only, the following parameter are indicated:

- Airspeed Indicator [Kts]
- Attitude Indicator [Ft]
- Fuel Indicator
- Turn And Bank Indicator
- Magnetic Compass.

The parameter mentioned are shown in the Fig 1.25 They are exactly how they will look like in the simulation:

Figure 1 .25

They do not like the instruments present in NH90. In any case, they are very useful diuring the simulation.

1.1.4 Motion platform

In order to make the simulation more realistic and to reduce the simulation sickness, the simulator will be built on a motion platform.

In the designed layout, there are two places, one for the pilot and other for the crew station, like in the Fig 1.26.

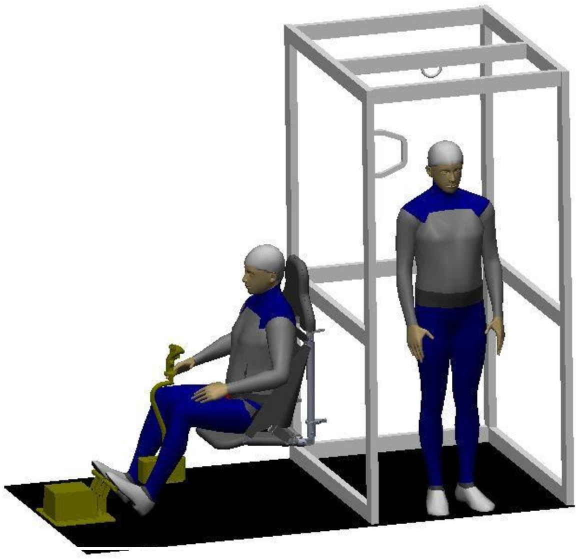


Figure 1.26 – Motion Platform and trainees position 3D sketch

Observing the Fig 1.26 it is possible to understand that the person sitting will be the pilot. The pilot will have an HMD device. The other person will stay in crew station and will have also the opportunity to sit down. An HMD device will be available also for the crew station.

This CAD is useful to understand the position of the trainee.

In the Fig 1.27 is present a CAD with the moving platform. The moving platform is on the bottom of the figure.

Figure 1.27

The moving platform is designed by Brunner Elektronik AG is specialized:

- Developing of simulation product
- Developing of drives and controls.

The moving platform used in this simulation is 6DOF MOTION 1000. The motion control system provides different motion cueing algorithms, allowing the motion system to be adjusted for different types of simulations.

It comes with an external Motion Control Unit including Power Electronics an embedded PC which is hosting and running the motion platform server.

The motion platform server communicates with the external environment via the UDP Network Protocol.

The external communication with the simulator interface is done over Gigabit Ethernet (In computer networking, Gigabit Ethernet GbE or 1 GigE is a term describing various technologies for transmitting Ethernet frames at a rate of a gigabit per second). A powerful Motion Configuration tool is used for downloading the Simulation Profiles, Setup of Washout filters and other parameters. (BRUNNER Elektronik AG)

In the Fig 1.28 the software interface is shown:

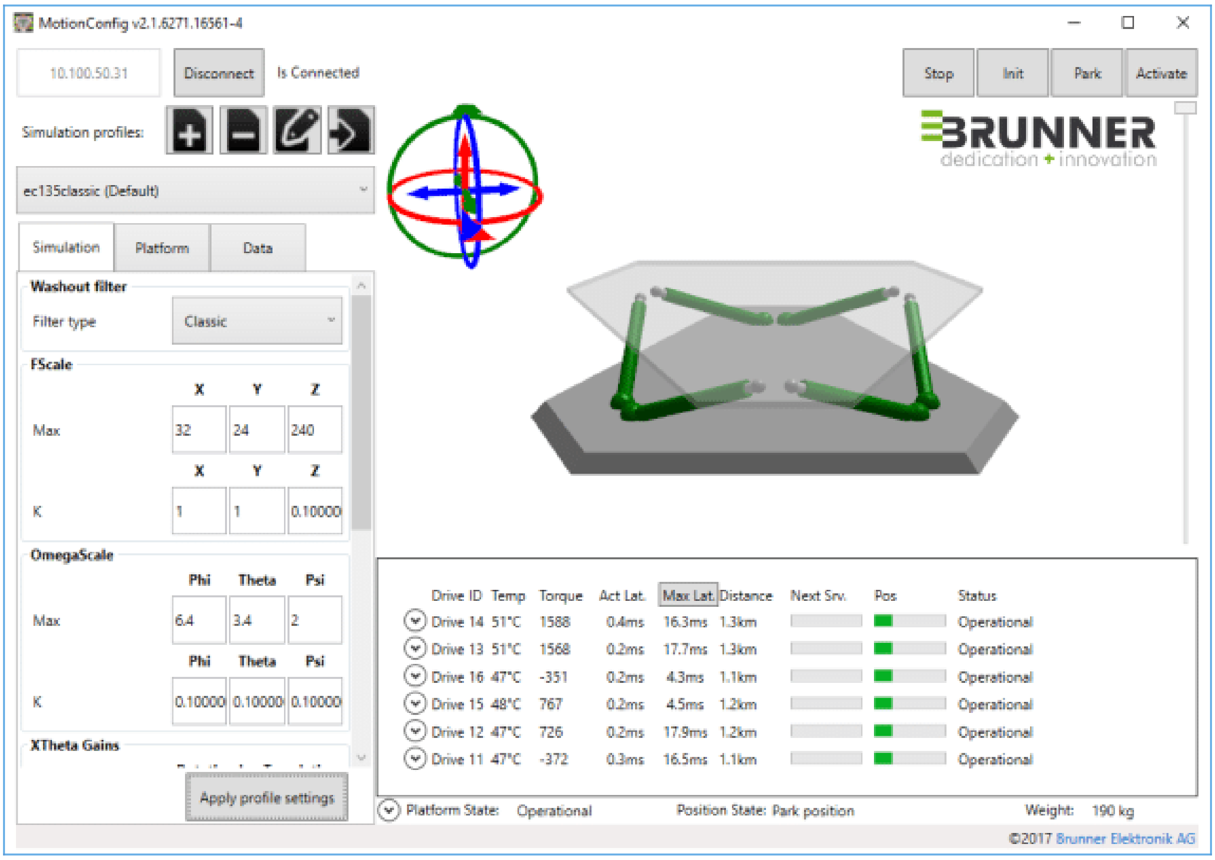


Figure 1.28 (BRUNNER Elektronik AG)

The most important features of the motion platform are: (BRUNNER Elektronik AG)

- High-fidelity motion cueing and tuning algorithms for maximum performance level.
- Safety architecture, including an integrated ‘Return-to-Home’ function to handle critical failure situations and two emergency-off switches.

- BRUNNER 6-DOF Motion Control Unit, hosting an embedded PC, which is running the intelligent motion platform server.
- Compact size and single phase mains power.
- Unique Price-to-Performance ratio.
- Application Interface via Gigabit Ethernet.
- Cost efficient Systems Design, ensuring maximum durability and lowest life-cycle-costs
- Signal Tower, indicating Systems Status

The platform moves thanks to hydraulic piston powered by an electric motor.
There are six hydraulic pistons and six electric motors.

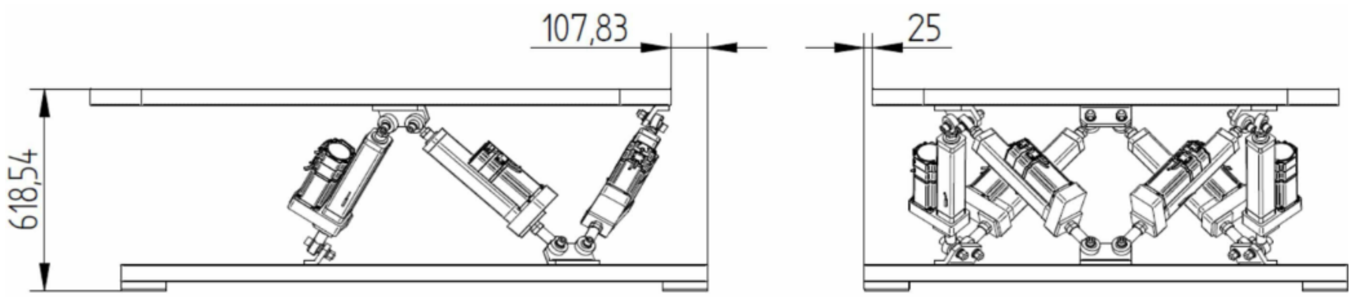


Figure 1.29 (BRUNNER Elektronik AG)

In the Fig 1.29 the hydraulic pistons and electric motors are shown, furthermore with the dimensions.

The specifications are: (BRUNNER Elektronik AG)

- Have (disp/velocity/acceleration): ± 125 [mm], ± 250 [$\frac{mm}{s}$], ± 4 [$\frac{m}{s^2}$]
- Pitch (disp/velocity/acceleration): ± 15 [°], ± 30 [$\frac{°}{s}$], ± 250 [$\frac{°}{s^2}$]

- Roll (disp/velocity/acceleration): ± 15 [°], ± 30 $[\frac{°}{s}]$, ± 250 $[\frac{°}{s^2}]$
- Yaw (disp/velocity/acceleration): ± 16 [°], ± 30 $[\frac{°}{s}]$, ± 250 $[\frac{°}{s^2}]$
- Surge (disp/velocity/acceleration): ± 140 [mm], ± 280 $[\frac{mm}{s}]$, ± 3 $[\frac{m}{s^2}]$
- Payload max: 1000 [Kg]
- Connection Type: 10/100/1000BASE-T
- System Weight: 250 [Kg]

In Fig 1.30 the external dimensions of the moving platform are shown:

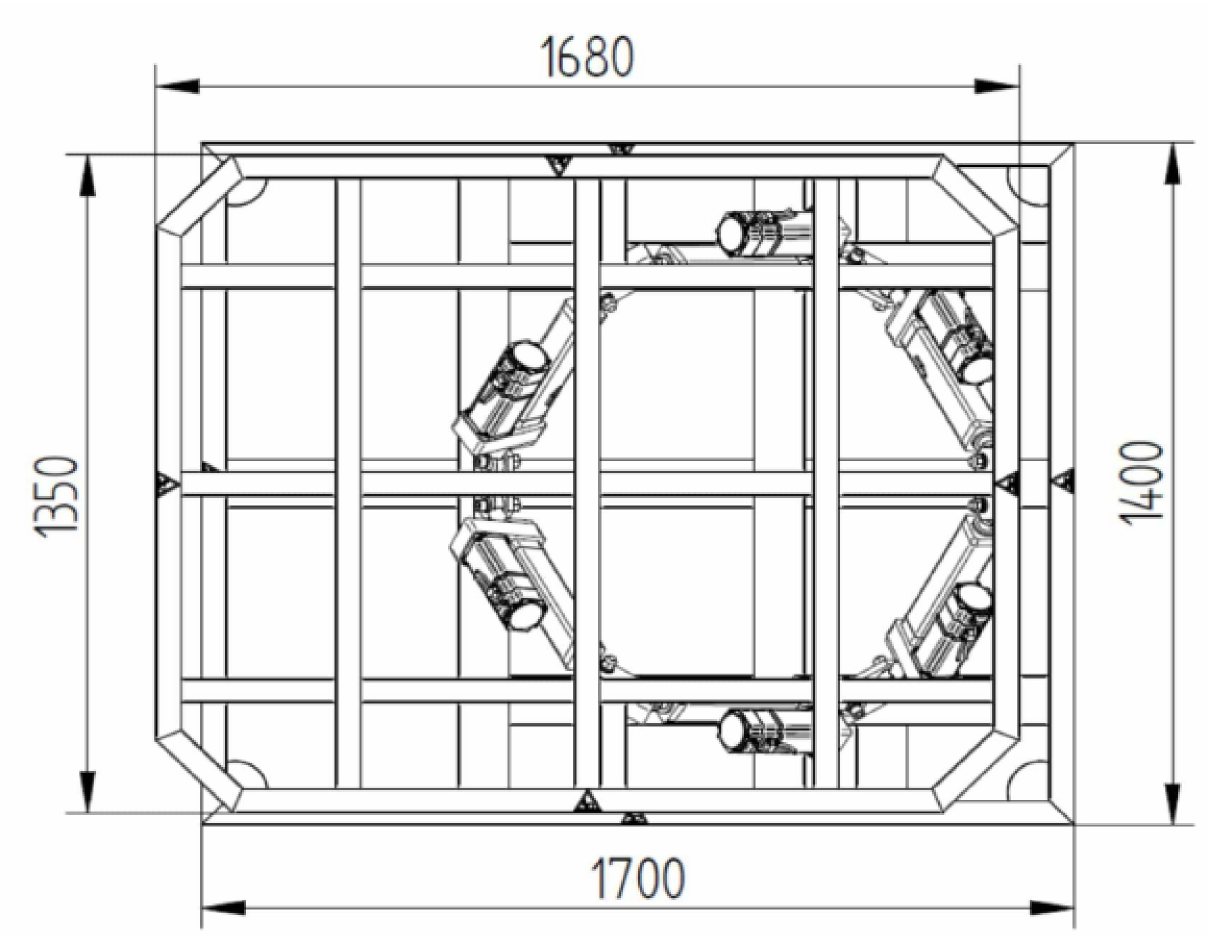


Figure 1.30 (BRUNNER Elektronik AG)

Above the moving platform the following devices will be installed:

- Metal racking. (Fig 1.24)
- Sit for the pilot.
- Sit for the crew station.
- Helicopter controls.
- Crew station control.

When all these devices will be implemented, it is possible to decide the position of the moving platform.

The motion platform with all device implemented will have another center of gravity. (Comparing with the center of gravity of the motion platform without implementation). In order to calculate the center of gravity's position is necessary to know the weight of each component.

In the Fig. 1.31 is shown the motion platform, the component present on the right part of the figure is the contention and software box. This component has to be placed on the laboratory's flor. There are also two red security buttons.

These buttons are useful if the moving platform doesn't respond to the commands.

The box has to be placed far away to the motion platform and in a safe place.

Figure 1.31 (BRUNNER Elektronik AG)

A motion platform will be a great addition for the fidelity of the flight simulator, and it will reduce the discrepancy between the vestibular and the visual system during the virtual reality simulation; that in flight-simulators is on the most sickening factors.

1.2 Immersive Flight Simulators

There are many other simulations in the word for the helicopter, in this paragraph, there is a brief comparison whit the other simulations.

Specifically, it is useful comparing the simulator devices.

The motion platform used for this simulation has the place for two trainees. This particularity is very important and very useful.

Usually, the existing simulations have been set only for the pilot trainer.

Recapitulating, the simulation made has the possibility to train in the following position:

- Pilot
- Crew station

The advantages for the trainees are:

- Simulate a real experience with a colleague in the same helicopter.
- Solve different tasks synchronized each other.
- The two trainers can communicate each other tanks to the specific communication devices.
- The trainers have a different view during the simulation, the pilot the front view and the crew station the right view.
- Possibility to use the simulation alone. In case of the trainer in the crew station, the helicopter will be autopilot.
- Works together is better because if one trainee does something wrong, the simulation of the other trainer will be compromised by this.

It is the most important and useful difference between the other existing simulations, the new implementation is the crew station. With this innovation, a military will have the possibility to trainer inside the helicopter. It is a good region for safe money and time.

In order to train in the crew station (without a simulation), there are many operations to do:

- fly with the helicopter
- Find a safe place where it is possible to train in the crew station, a big place without civil or houses.
- Organize the specific task, it makes a lot of time, with the simulation is very easy to change the tasks.

In the end, the trainer has the opportunity to train often before entering a real helicopter. In order to understand better is useful to have a brief explanation of the other simulation.

1.2.1 VMS -- Vertical Motion Simulator (NASA):

The VMS offers an unequalled range of motion, moving as much as 60 [ft] vertically and 40 [ft] horizontally. This is key to high-fidelity simulation, and along with other sensory cues, makes the VMS unsurpassed at simulating aerospace vehicles for the entire flight envelope, especially during the critical phases of approach and landing. The Space Shuttle Program continues to reduce program risk by leveraging the VMS capabilities for engineering studies, design validation, and astronaut training (George L. Danek, 1993). First, engineers can customize the system to simulate any aerospace vehicle, whether existing or in the design stage. Second, simulations occur with high fidelity; that is, the simulator reproduces the flight characteristics of the vehicle with a high degree of accuracy. This entails delivering realistic cues to the pilot in real time.

In the Fig 1.32 the Vertical Motion Simulator is shown:

Figure 1.32 (George L. Danek, 1993)

The main difference between the moving platform used in this work is:

The Vertical Motion Simulator requirement a lot space and it is not possible to move it.

With the moving platform used in this work, it is possible.

The Vertical Motion Simulator requested a specific room with particular devices like in the Fig 1.33

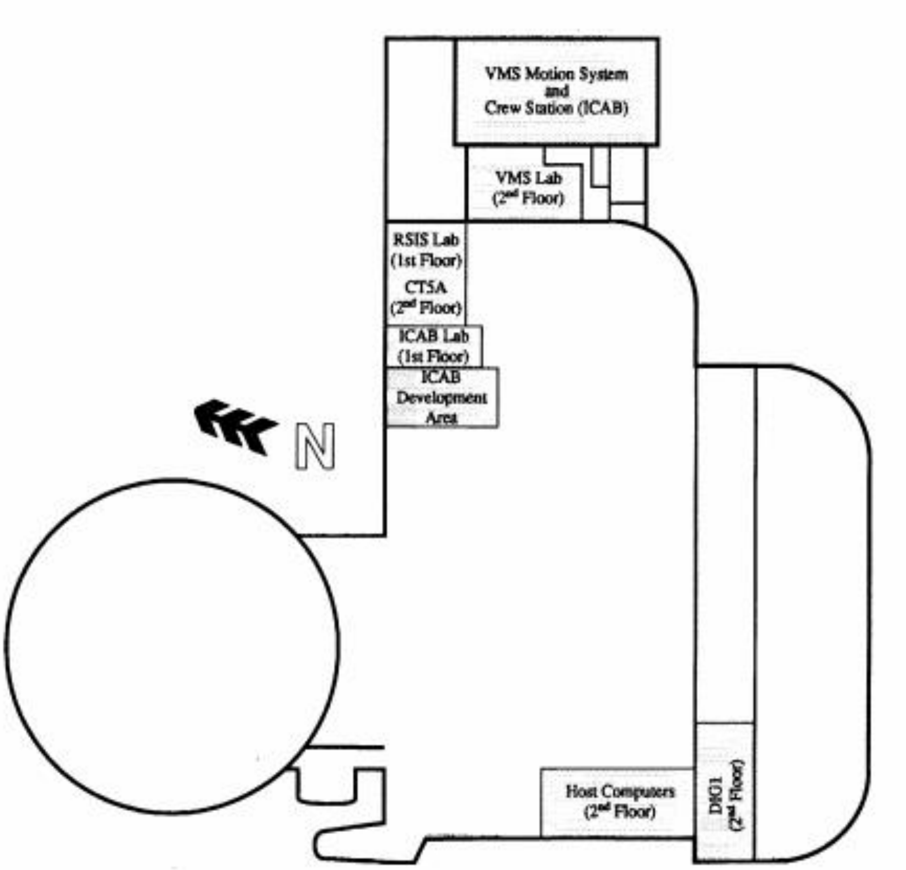


Figure 1.33 (George L. Danek, 1993)

1.2.2 Desdemona

Desdemona has the capability to support the outside view with exactly the right movements, in order to generate the desired driver or pilot behaviour. This is called motion cueing.
Desdemona combines the possibilities of both the hexapod and the human centrifuge. The cabin has a modular layout that allows to be used as a fighter jet, a helicopter, a Boeing 737, a spacecraft or - and just as easily - as the interior of a car or the bridge of a small ship. It is mounted on a fully gimbaled system that is able to rotate around any conceivable axis. (Dr. W.Bles et al.). The system as a whole allows 2 [m] of vertical movement, combined with 8 [m] along with a horizontal sledge. The sledge itself is able to spin as well. Centrifugation enables Desdemona to generate constant G-forces up to a maximum of 3 G.

The main applications are:

- Spatial disorientation.
- Military fly simulation.
- Motion cueing.
- Research on human movement.
- The human factor in mission simulation (Networking Enables Capabilities).

In the Fig 1.34 the Desdemona is shown:

Figure 1.34 (Dr. W.Bles et al.)

1.2.3ACME NH90 Dynamic Motion Seat

ACME's NH90 helicopter dynamic motion seats are a combination of vehicle crew seats with electric motion components built inside the seat. The seats look, feel, and function like actual crew seats, and provide cues to emulate sensations felt during the vehicle or in-flight operations.

ACME seats translate simulator acceleration signals into realistic, convincing motion. The seats provide a higher level of cueing, where the motion IS training, for example feeling unique signature sensations such as translation lift between helicopters vertical and horizontal flight or the difference between a helicopter rotor out of balance or a blade of the track. (www.acme-worldwide.com)

The seats use a patented electric system with individual motion plates/pans, providing cues directly to the crew's body. It's the effect of multiple channels in the seat working seamlessly together that provides complete, realistic, immersive cueing.

Also, in this case, the seat is available only for the pilot position.

It is the official motion seat of NH90, the motion platform used in this work it will be the future development of the Dynamic Motion Seat.

In the Fig 1.36, Fig 1.37 and Fig 1.38 there are a top view, bottom view and the headrest of the Dynamic Motion Seat.

Figure 1.36 (www.acme-worldwide.com)

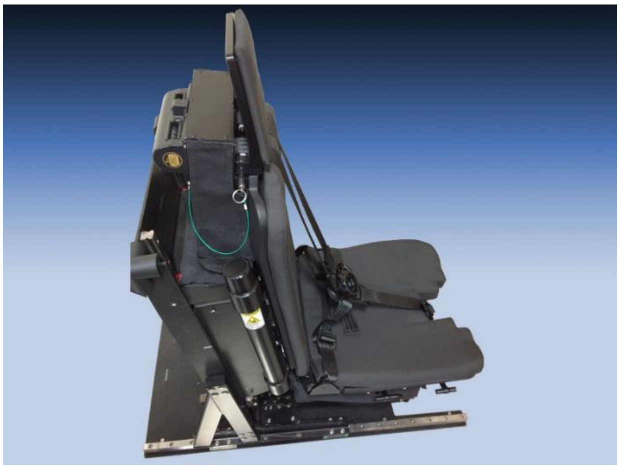


Figure 1.37

Figure 1.38 (www.acme-worldwide.com)

1.2.4 BEC Motion Simulation

The BEC motion simulator is based on commercial six-axes serial robots from KUKA, originally designed for use in industries as a manipulator. (www.motion-simulators.com)

There are many possibilities like:

- Higher dexterity
- Larger motion envelopes
- Sustained centrifugal accelerations
- To actually place pilots in extreme orientation

The main characteristics are (www.motion-simulators.com)

- Highly versatile simulation based on a multiple use carbon fibre.
- A system with the infinite rotational base axis.
- Additional axis for dynamic updating mounting in real-time.
- Enlarged working space with a robotic simulator mounted on a dynamic linear rail.
- Biggest robotic simulator for series identical cabins up to a weight of 850 [Kg].
- Mobile simulator integrate.

There are different types of this simulator; the main characteristics mentioned above are generally for all types of the simulator.

In the Fig 1.39, a conceptual scheme of the simulator is shown

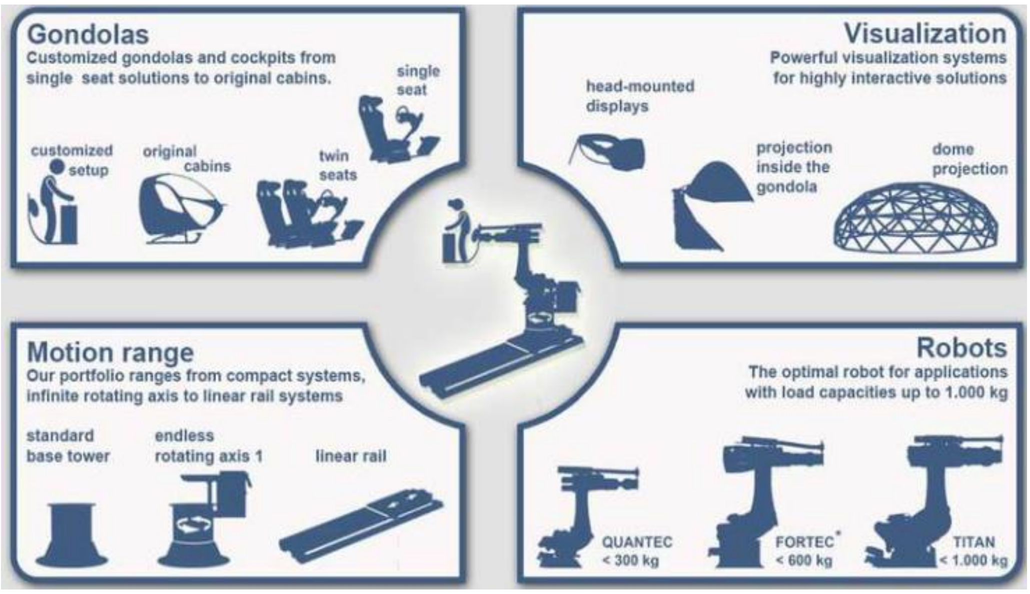


Figure 1.39 (www.motion-simulators.com)

The BEC Motion is shown in Fig 1.40, (BEC 6DOF)

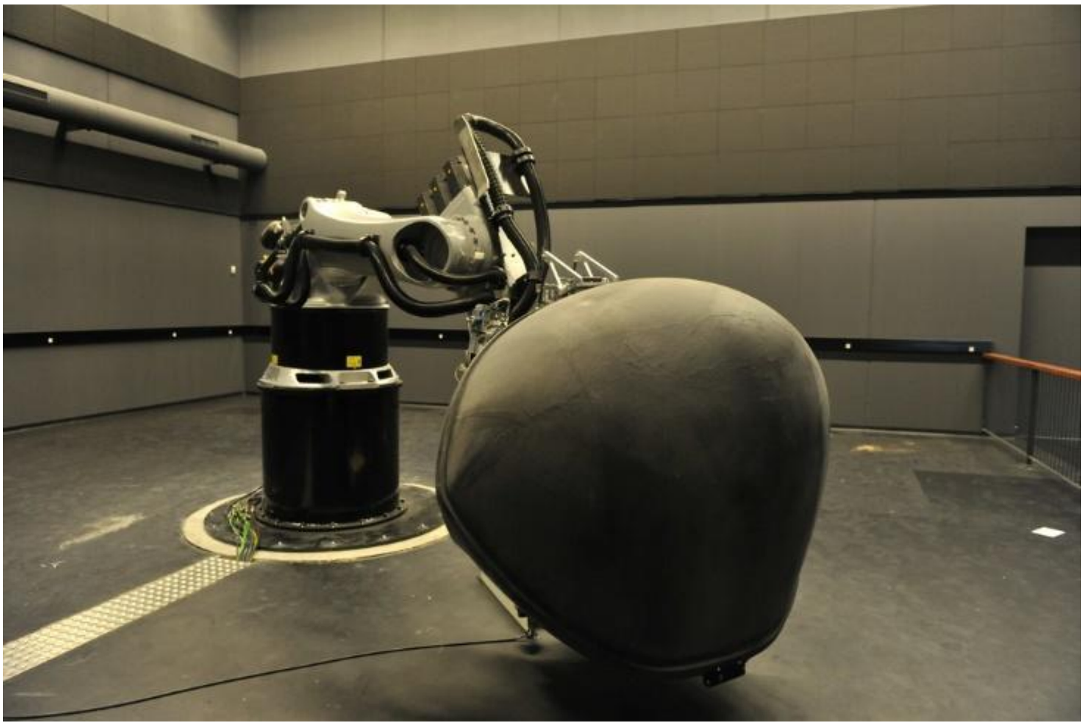


Figure 1.40 (www.motion-simulators.com)]

1.2.5 UH-60 Black Hawk Flight Simulator

It is a motion platform with a cab implemented. The cab is equipped with two seats one for the pilot and one for the co-pilot. Furthermore, there are two monitors (no HMD device).

The dimensions of the simulator are (TM 55-1526-237 Series):

- Cockpit area: 480”× 516” ”× 324”
- Computer area: 480”× 360” ”× 120”
- Hydraulic system: 144”× 192 ”× 120”

This simulator is used for the helicopter and aircraft.

To provide training in the techniques of day/night visual and instrument flight and to aid in maintaining proficiency in these techniques after completion of formal training. The trainer is used for initial and refresher training of aviators in cockpit procedures, emergency procedures, day/night visual and instrument flying techniques. (TM 55-1526-237 Series)

It is impossible to carry the simulator. The simulator is shown in the Fig 1.41

Figure 1.41 (TM 55-1526-237 Series)

1.3 Human Factors – Simulation Sickness

Simulation Sickness represents an important problem with virtual reality environments. Simulation Sickness present similar symptoms to motion sickness but the cause is distinct; motion sickness occur when the subject is stationary but has a sense of motion induced through exposure to changing visual imagery. (Joseph J et al.,2000)

Typical symptoms of simulation sickness are:

- Salivation
- Sweating
- Nausea
- Stomach awareness
- Nausea
- Fatigue
- Headache
- Eyestrain
- Difficulty focusing.
- Vertigo

Simulation sickness can be caused by different factors, the main three theories as the cause of simulation sickness are:

- Sensory Conflict Theory
- Poison Theory
- Postural Instability Theory

1.3.1 The Vestibular and the Visual System

Before discussing the theories about the occurrence of simulation sickness and how to mitigate it, it is useful to understand the relation between that relate to self-motion, the vestibular system and visual perception.

<u>Vestibular System</u>

The Vestibular system (Fig. 1.42), is a system located inside the non-acoustic part of human's inner ears. It provides information about the movement and orientation of the head in space (Joseph J et al.,2000). The three semicircular canals that can be seen in Fig 1.39 detect the linear acceleration in the three dimensions.

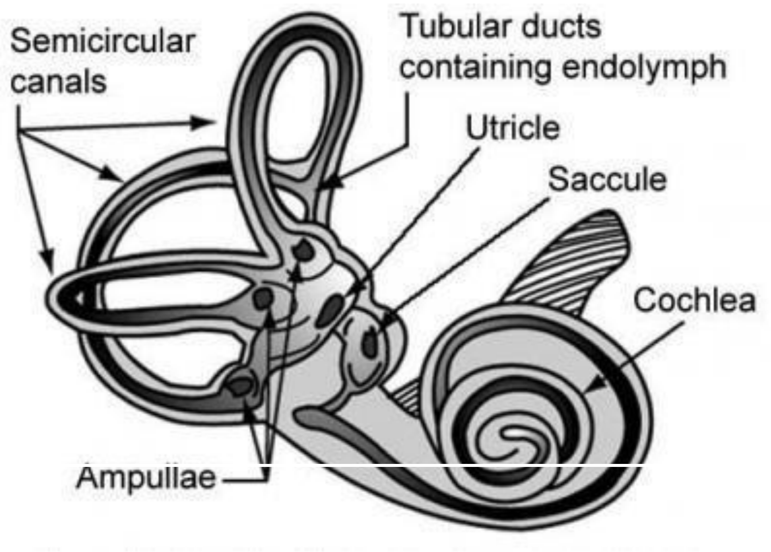


Figure 1.42

The canals are filled with a fluid, as the head has an angular acceleration the fluid flows through the canal and deflects cells, wich send signals to the receiving areas of the brain.

It is important to know that there are two vestibular components and one mirror the other; if one side of the head push the other should pull, if both of them do the same thing vertigo will result.

The utricle and saccule detect the linear acceleration, but their task is to provide vertical orientation with respect to gravity.

Visual Perception of Self-Motion

A basic concept to know before discussing this topic is about the vection. The vection is a phenomenon that can occur for example when a person is sitting on a stationary train and the adjacent one begins to move.

In standard self-motion the motion would be accompanied by vestibular information, but with vection, the vestibular information is not present. This happen even in virtual Enviroments.

On this fact is based the sensory conflict theory that will be explained in the next paragraph.

Another factor that causes is the apparent depth of the objects in the virtual environment. Furthermore, in flight simulators where motion makes part of the application, it will provide more stimulus that induce vection.

Relation between the Vestibular system and Visual System

The vestibular system keep the eyes in position when the head moves. Each semicircular canal interacts with a single eye muscle pair. If the head moves to the left and the person is looking at a stationary point, the left horizontal canal of the vestibular system will send messages to the right lateral rectus in order to pull the eyes to the right (Fig 1.43). Vice versa if the head turns right the vestibular system will interact with the eyes' muscles in order to pull them left.

This phenomenon is known as the vestibulo-occular reflex.

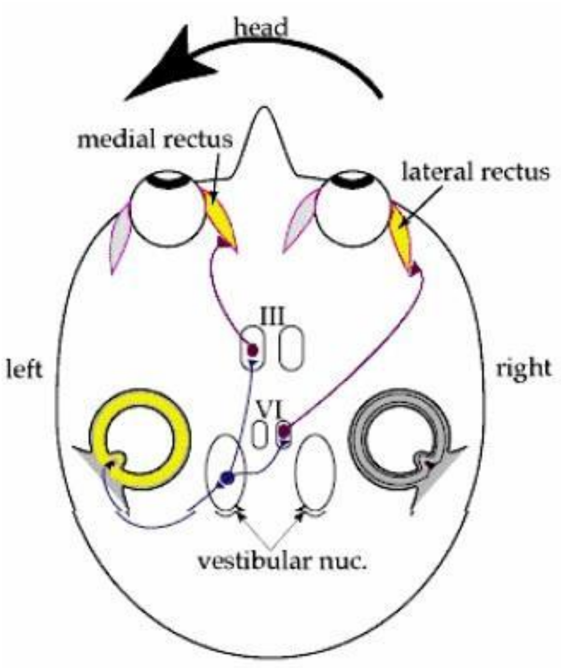


Figure 1.43

1.3.2 Simulation Sickness Theories

The three theories that discuss on how simulation sickness occurs are explained below.

<u>Sensory conflict theory</u>

The sensory conflict theory is the most recognized theory about the simulation sickness occurrence.

The sensory conflict starts when the subject experience a discrepancy between the vestibular and the visual system.

For example, in a virtual reality flight simulator the subject will experience a sense of motion by his visual system but his vestibular system will not send any message of movement. When this conflict arise, simulation sickness may occur.

Poison Theory

This theory is a possible explanation of the simulation sickness from an evolutionary point of view. It compares the effect that a subject would experience when being poisoned with the effect that virtual environments have to a subject.
A poison could provoke a discrepancy about the coordination of the visual and the vestibular system; the physiological response of a human body would be removing the content of the stomach.
Anyway, this theory does not explain why people who experience cyber sickness do not always have an emetic response.

Postural instability theory

The postural stability is the state in which uncontrolled movements of the perception and action system are minimized (Gary E. Riccio e al., 1991).The postural stability theory is based on the idea that, in order to not feeling sick, humans have to maintain postural stability in the environment.
Depending on the environment the postural stability will have different patterns: for example, a person walking on ice or walking on concrete will assume a different postural stability in order not to fall.
If the surrounding environment changes in an unexpected way, postural control will be lost, and a state of postural instability will arise.

Using virtual reality. the surrounding environment suddenly changes, therefore it provokes postural instability. The postural instability theory states that the cause of motion sickness is prolonged postural instability (Gary E. Riccio e al., 1991) and it increase with the duration of the instability.

1.3.3 Contributing factors in Virtual reality

There are several factors that are sickening when using Virtual Reality.
These factors can be divided in two categories:

- Technological issues
- Individual Factors

Technological issues

The technology used to stay in a virtual environment like HMDs has some limitations that can cause simulation sickness. The factors that are recognized to cause simulation sickness are:

- Head Tracking Error: A great problem using HMD is to have a good head tracking when someone is in an virtual environment. Errors during in the head tracking can cause a latency between the head movement or a non-perfect orientation of the virtual image with the eyes.
- Lag: when the virtual environment has to represent an elaborated scenario it is possible that arise a delay between the movements in the virtual word and the real word.

- Flicker: when the hardware and the configuration of the devices are not accurate or the field of view is too wide, flickering will be distracting and tiring for the eyes.

Individual Factors

These factors are being considered in order to explain why some people is sensitive to simulation sickness and others not.
The main Individual Factors are:

Age: Studies state that the susceptibility simulation sickness varies with the age and increases for the youngers, over 50 years old the simulation sickness is lower (J.T. Reason et al.,1975).

- Position in the simulator: People sitting in the simulator are the ones that experienced the lower simulation sickness.
- Experience with virtual environments: People that is new with virtual reality tend to fell sicker with virtual reality simulations.

1.3.4 Evaluating Simulation Sickness

In order to evaluate the intensity of simulation sickness, psychologists often provide to trainees a questionnaire like the one in Fig. 1.44 to be filled in after the virtual reality exposure.

No_____________ Date__________________

SIMULATOR SICKNESS QUESTIONNAIRE
Kennedy, Lane, Berbaum, & Lilienthal

Instructions : Circle how much each symptom below is affecting you right now.

1. General discomfort	None	Slight	Moderate	Severe
2. Fatigue	None	Slight	Moderate	Severe
3. Headache	None	Slight	Moderate	Severe
4. Eye strain	None	Slight	Moderate	Severe
5. Difficulty focusing	None	Slight	Moderate	Severe
6. Salivation increasing	None	Slight	Moderate	Severe
7. Sweating	None	Slight	Moderate	Severe
8. Nausea	None	Slight	Moderate	Severe
9. Difficulty concentrating	None	Slight	Moderate	Severe
10. « Fullness of the Head »	None	Slight	Moderate	Severe
11. Blurred vision	None	Slight	Moderate	Severe
12. Dizziness with eyes open	None	Slight	Moderate	Severe
13. Dizziness with eyes closed	None	Slight	Moderate	Severe
14. *Vertigo	None	Slight	Moderate	Severe
15. **Stomach awareness	None	Slight	Moderate	Severe
16. Burping	None	Slight	Moderate	Severe

Figure 1.44 (Kennedy et al.,1993)

CHAPTER 2

VBS3 Simulation and Visual Stimuli Logging

2.0 Introduction

VBS3 is a fully interactive, 3D training system that provides a premium synthetic environment suitable for a wide range of military (or similar) training and experimentation purposes. Supporting rapid terrain development, VBS3 offers large terrain areas with an unparalleled level of environmental realism and supports realistic modeling of characters, vehicles, and equipment. (bisimulations.com)

VBS3 is a training tool including after-action review, HLA / DIS compliance, and a comprehensive mission editor that allows any imaginable scenario to be created and also modified in real time.

In the lines before the HLA and DIS have been mentioned, the meanings are:

- DIS: Distributed Interactive Simulation (DIS) is an IEEE (Institute of Electrical and Electronics Engineers) standard for conducting real-time platform-level wargaming across multiple host computers and is used worldwide, especially by military organizations but also by other agencies such as those involved in space exploration and medicine. (Mark McCall et al.,2010)
- HLA: High-level architecture (HLA) is a general purpose architecture for distributed computer simulation systems. Using HLA, computer simulations can interact (that is to communicate data and to synchronize actions) with other computer simulation regardless of the computing platforms. The interaction between simulations is managed by a run-time infrastructure (RTI). HLA is an interoperability standard for distributed simulation used to support analysis, engineering, and training in a number of different domains in both military and

civilian applications and is the standard technical architecture for all US Department of Defence simulations. (Francesca G. et al., 2012)

There are others two versions of VBS: VBS1 and VBS3, VBS3 is the most recent and upgrade.

VBS3 is developed by Bohemia interactive simulation.

Bohemia interactive simulation began as spinoff studio Bohemia Interactive Australia (BIA), formed by Bohemia Interactive Studio and David Lagettie, where the joint development of a special military training simulation program VBS1 or Virtual Battlespace 1 began in December 2001. (bisimulations.com). The Virtual Battlespace software series is widely used as a desktop simulation software for training among western military organizations. The company's customers include the United States Army, the United States Marine Corps, the United Kingdom Ministry of Defence, NATO, the Australian Defence Force, the Canadian Armed Forces and the Swedish Armed Forces. Today, the company has offices in the United States, United Kingdom, Poland, Czech Republic and Australia.

They have developed the following software :

- Arma 3
- Arma2
- Arma1
- Arma Tactics
- DayZ
- Take On Mars

Bohemia Interactive Simulation is proud to be members of the following industry organizations:

- National Defense Industrial Assistant.

- National Training and Simulation Association (NTSA)

Moreover, by signing the Covenant, Bohemia Interactive Simulation commits to ensuring members of the Armed Forces community face no disadvantages in business dealings, the company supports veterans of all ages working with the Career Transition Partnership, Bohemia interactive simulation support employees who choose to be members of Reserve forces and they support local cadet units, among other principles honouring service members.

Chapter from 2.1 to 2.5 focus on VBS3 simulator, the scenario and logic implementation start at chapter 2.6 “Training Scenarios”.

2.1 VBS3 Functions

First, it is necessary to create a specific profile; there are two different logins:

- Administrator: Whit this login it is possible to make a new scenario, change the mission, to manage the multiplayer simulation etc. In other words, with the administrator login, it is possible to do all operations that VBS3 allows.
- User: With this login, it is impossible to do any operation as well as playing. This login is very useful for the multiplayer section.

2.1.1 User

The first interface that will appear in the monitor is shown in the figure below.

*Figure 2.1 SEQ Figure_4 * ARABIC 1*

In the following lines, all different options will be explained

- Training scenario: Provides access to a set of guided single-player missions. Several missions are present. These missions are useful for familiarizing yourself with the operation of your character, equipment, and vehicles in VBS3. Each Training Mission focuses on a different skill; including basic movement, weapons training, vehicle training and navigation.
  In order to play with the missions the following steps have to be followed:

  1) Log in to VBS3 and access the Main Menu.
  2) Click Training scenario
  3) Select a Training Mission to view its description.
  4) Toggle Regular /Veteran (easy / hard, respectively).
  5) Click Start to view the mission briefing.

It is also possible to create a scenario and after that export it in the training scenario.

- Networking Menu: This option is used when you want to play with other people. Another computer has to connect; otherwise, this option does not work.
  In the figure below is shown the Networking page.

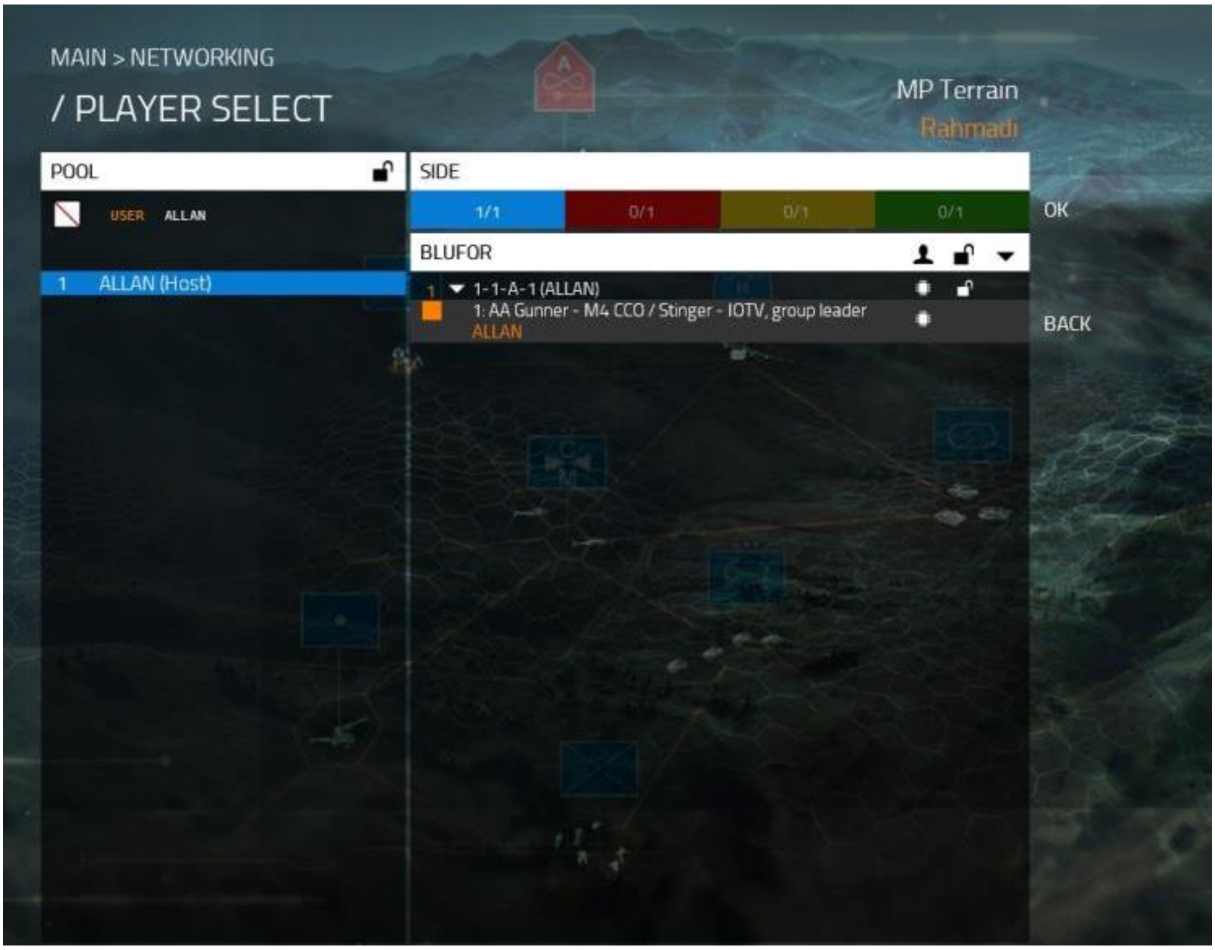


Figure 2. 2 (VBS3 Manuals,2017)

There are different colors on the top of the figure (Blue: Blufor, Red: Opfor, Yellow: Independent, Green: Civilian)

- After action menu: Provides access to the post-mission review tools.
- Library: The library (Fig. 2.3) is useful to familiarize with the weapons and vehicles in VBS3.

Figure 2.3 The library of Equipment and Vehicles in VBS3 (VBS3 Manuals,2017)

In the library all vehicles, units and objects will be shown. There is also the option to try the vehicle.

- Options: In the option menu several simulation characteristics can be set up:

1) Audio: The settings to control the sound in VBS3 are:

Effects: General sound effects, such as weapons fire, explosions and engine noises.

Radio: AI radio communications chatter volume.

Voice over network boost: The boost to apply to VOIP network communications from other users. Radio and effects volume setting affect the volume of VOIP transmissions.

UI Sounds: Menu selection sounds.

Subtitles: Select to display captions associated with simulation sounds.

Radio: Select to display the captions associated with the AI radio messages. (VBS3 Manuals,2017)

2) Controls: The Controls configuration enables to view and modify the controls for VBS3. It is shown in figure below.

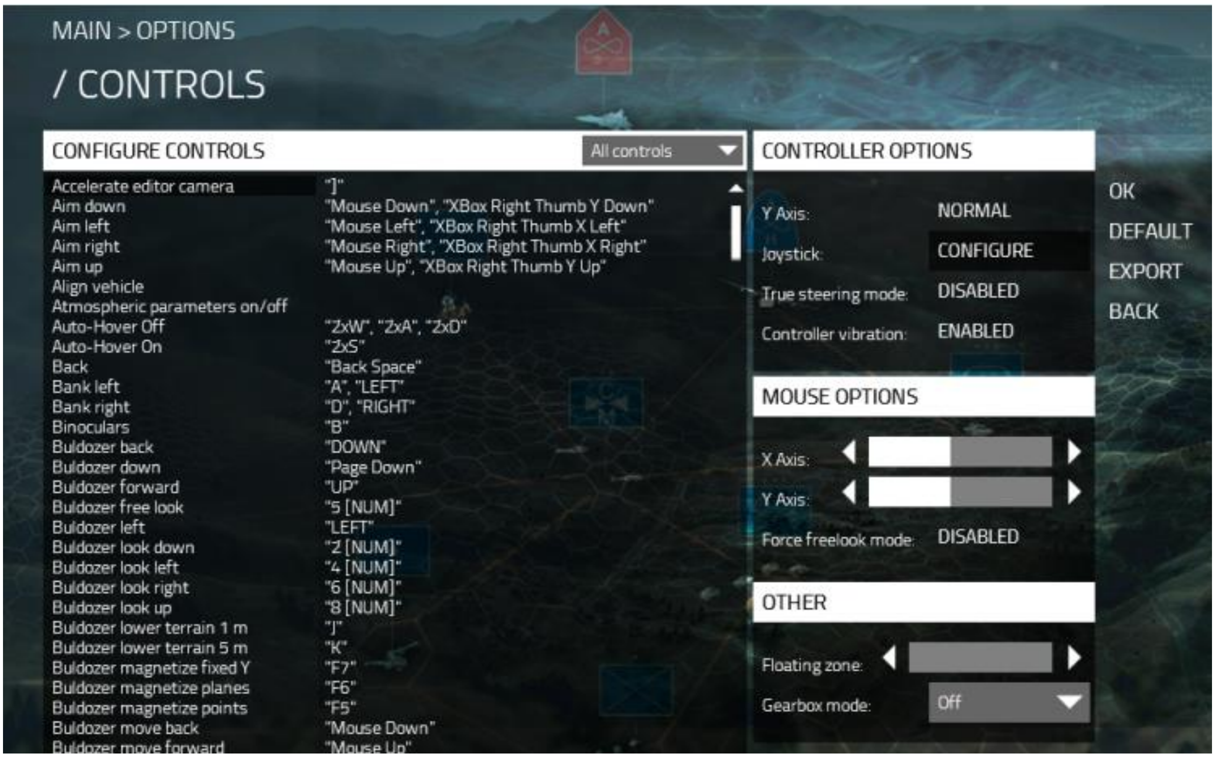


*Figure 2.4 SEQ Figure_4 * ARABIC 4 (VBS3 Manuals,2017)*

- Extensions: Provides access to plug-in management.
- Change profile: Returns to the User Login panel.

2.1.2 Character View

VBS offers a number of viewing perspectives.

The primary view in VBS is a first-person view display the perspective from the character viewpoint. This default view displays the health bar in the top left and a weapon status panel in the bottom right.

For instructional purposes, the administrator may enable a third person view, from behind and above your character.

It is possible to switch between first and third person view pressing the Numpad Enter command.

The team leaders can access a tactical view that provides a perspective of their entire team.

This view provides team leaders with an improved awareness of their immediate area and visibility of how their troops are positioned. This can be useful for assessing formation compliance and effectiveness in various situations.

In the Fig 2.5 and Fig 2.6 the first and the third person views 6 are shown.

Figure 2.5 First-person view (VBS3 Manuals,2017)

Figure 2.6 Third person view (VBS3 Manuals,2017)

2.1.3 Character Movement and Posture

In order to control the character, the mouse is used to turn the player's head, along with the following movement keys:

- WASD: Move forward, back, left, right
- Q: Lean left
- Q double-tap: Evasive roll to the left
- E: Lean Right
- E double-tap: Evasive roll to the right.
- Shift + W: Run
- W + W: Evasive forward (Sprint)
- RCtrl + W: Fast forward
- F: Jumps/step over/step onto
- Ctrl + Ctrl: Toggle weapon at ready position (raised)
- X: Crouch
- C: Stand
- Z: Prone

These are only the commands for the Units; in the following figures, the complete commands are shown.

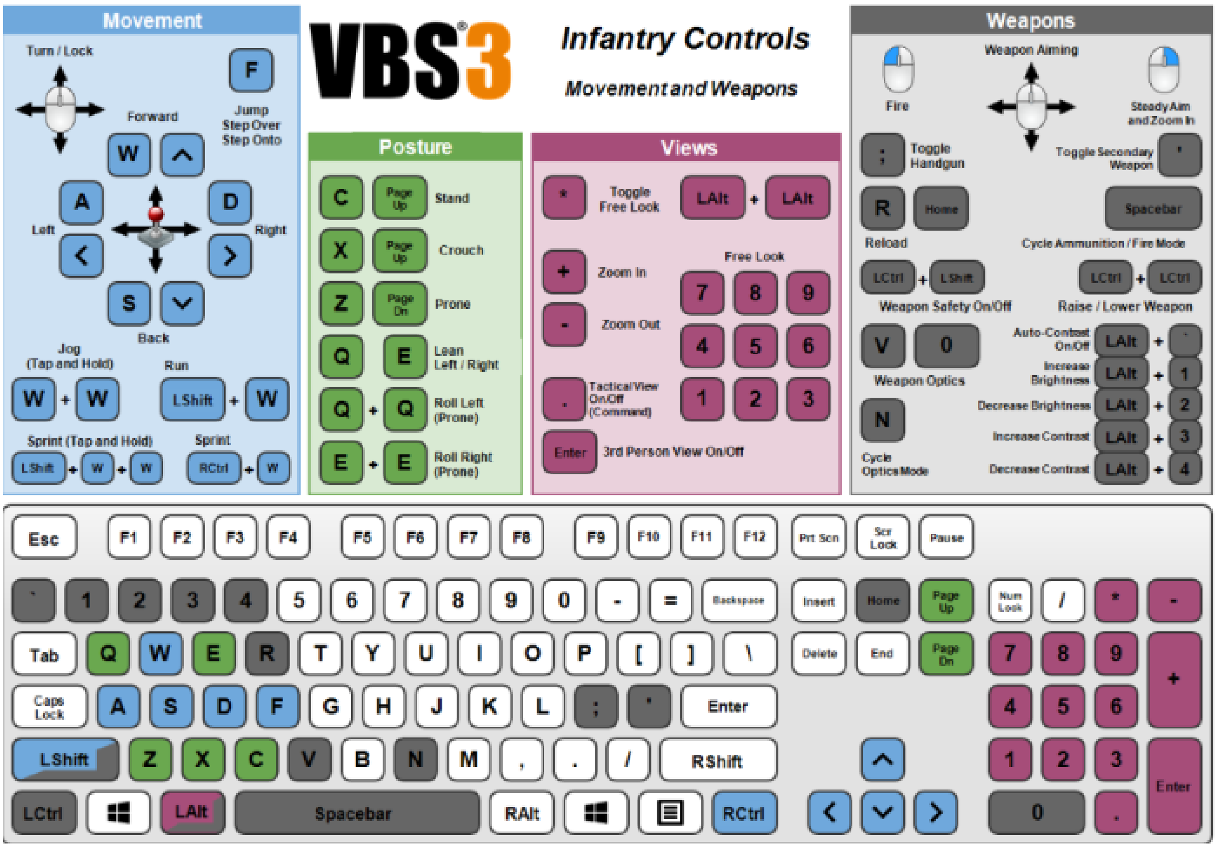


Figure 2.7 Movements and Waypoints (VBS3 Manuals,2017)

2.2 Action menu

The action menu is the primary method for interacting with a nearby object in VBS. Also, it is possible to add an action menu to the Unit (usually by script).

This command will be active by the scroll button mouse.

This menu is a context that contains the set of available actions for the following nearby objects:

- Personal equipment and weapons that it is possible to carry.
- Objects on the ground.
- Control stations, static weapons, vehicles and their equipment.
- Occupy vehicle, including specific equipment and actions for the specific position.
- Terrain obstacles, such as ladders and gates
- Other characters.

There are many operations available with the Administrator login, the main options will be explained in the successive pages.

Refereeing to the first interface of VBS, click on Mission Editor. It will appear a list of different scenarios. The list is shown below:

- As Samawah
- Baghdad, Green Zone
- Baghdad, Green Zone [16 Km]
- Geotypical Afghanistan [25Km]
- Geotypical Eastern Europe [25Km]
- Geotypicla Southwest USA [25Km]
- Geotypical Tropical [25Km]
- Greyland
- Open Sea 5Km
- Porto

- Prison
- Rahmadi
- Sahrani
- Takistan 128Km Multimap
- Takistan [13Km]
- Takistan [13Km]
- Warminster [5Km]

These scenarios are implemented in VBS. It is not possible to change or add a new one; It is only permitted to operate inside.

The specific scenario has been chosen and after the editor interface, it will appear.

There are two different views for the editor interface; 2D and 3D. Both of them are shown in Fig 2.40 and Fig 2.41.

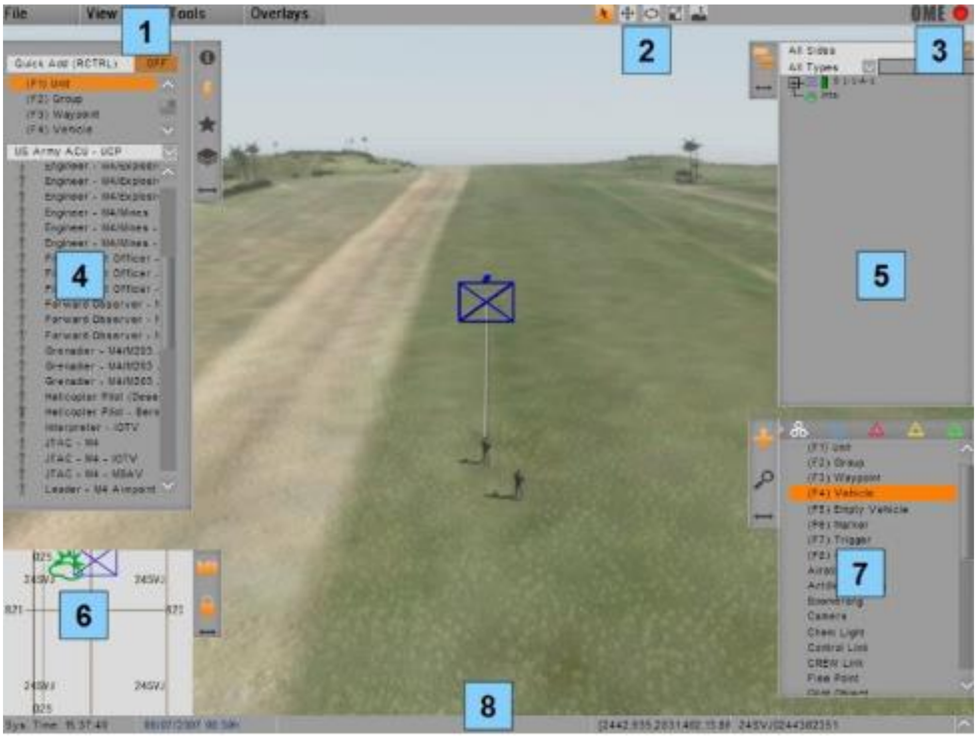


Figure 2.40 3D View (VBS3 Manuals,2017)

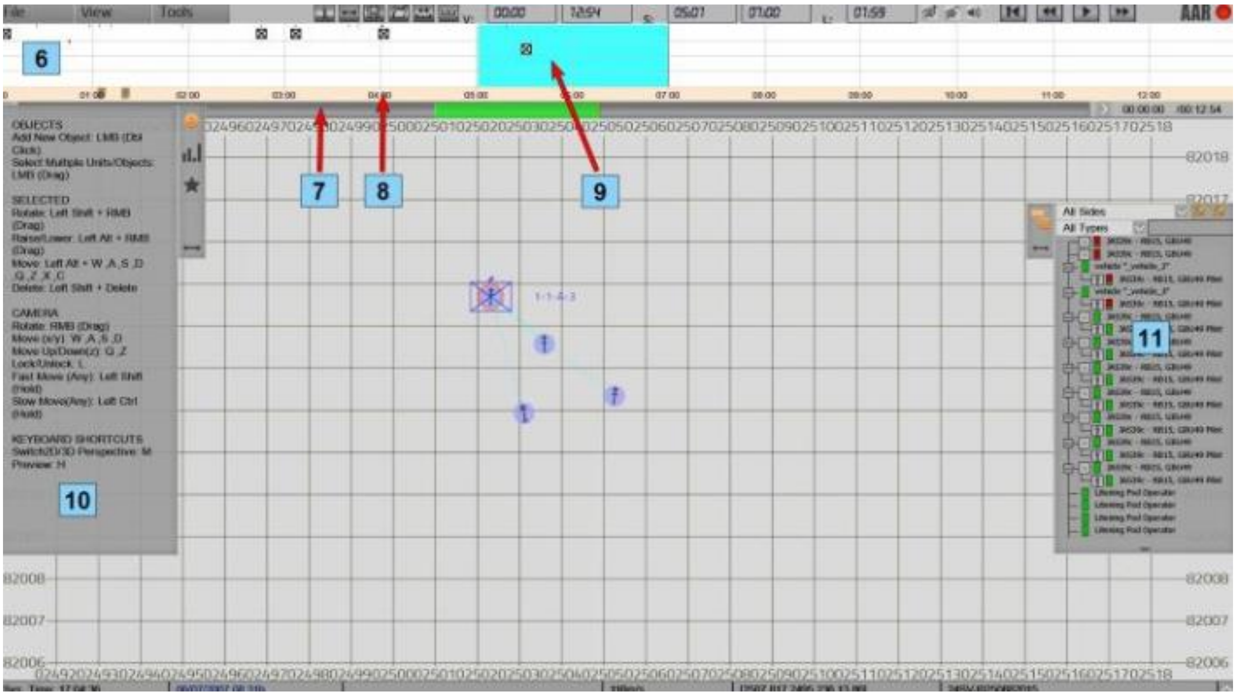


Figure 2.41 2D View (VBS3 Manuals,2017)

2.3 Unit

Every scenario requires at least one playable unit in order to start. The 'player' unit is the unit that is controlled by the user during the scenario preview (there can be only one player unit per scenario). 'Playable' units are those that are available to be controlled by human participants in a network session and there can be any number of playable units.

In order to add a new unit, click with the right bottom of the mouse and select “add new unit”.

Once a unit has been placed on the terrain, it can be visually identified as either a Player, Playable or AI controlled unit. A Player unit will have a red halo, a Playable unit will have a black halo and AI control with no Halo. Like in the figure below.

Figure 2.42 (VBS3 Manuals,2017)

In the Fig 2.43, the interface with the user is shown:

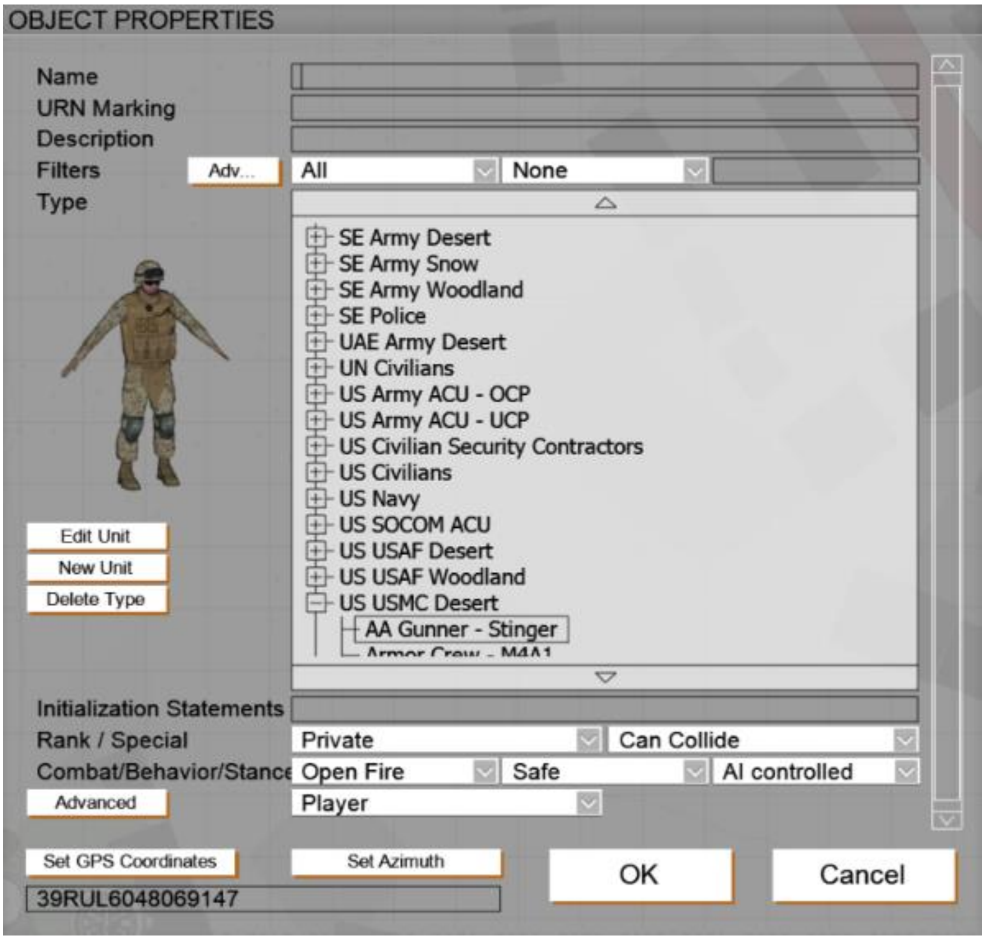


Figure 2.43 (VBS3 Manuals,2017)

There is also the possibility to create a Group of units, A group is a collection of soldiers with a common leader. Groups can be hierarchical.

The simplest method of adding a group is via the Add Group dialog. To view the Add Group dialog, left click on Group in the Editor Object List panel and then double click on the 2D map or 3D view where you want to place the group.

In the group option the bulfor, opfor, indipendet and civilian are available.

It is also possible to set the playability status for the group:

- Non playable: The group will be entirely controlled by AI
- Playable: The entire Group will be playable (each unit will be created as playable)
- Player as commander: The groups' leader will be the player unit, all subordinates will be non-playable AI
- Playable, Playable as commander: The group's leader will be the player unit, all subordinates will be playable AI

2.3.1 Groups

Group Editor option is available in the Add Group dialog. It is possible to define new types of groups that will be saved for later use. To access the group editor click on the Group Editor button.

Groups are sorted by 'category' and 'type'. Using the group editor it is possible to define new group categories and types.

There are five buttons on the left under Members allow that it is possible to move unit or vehicle types from the Available tree to the Members list.

The unit or vehicle at the top of the list will automatically be assigned as the leader of the group.

2.3.2 Vehicles

Basically, it is the same for the units, but many options are different.

It is possible to create a playable vehicle or to add crewed or empty vehicles by selecting the applicable entry in the Has Crew combo box. A crewed vehicle will automatically have AI entities manning the driver, gunner and commander positions within the vehicle. The vehicle's crew will appear as new units in the Editor Object tree.

The Playable combo box serves the same function as the Player and Playable combo boxes on the Add Unit dialog. The following options are available:

- Non-Playable: The vehicle will be entirely controlled by AI
- Player: The vehicle has only one playable crew position for the player character7
- Playable: The vehicle has only one playable crew position for the player by a human participant in a network session.
- Player as Pilot/Driver/Gunner/Commander: The player character will occupy the selected crew position in this vehicle
- Playable as Pilot/Driver/Gunner/Commander: The specified position(s) will be playable by a human participant in a network session.

2.4 Objects

Objects within the mission editor are 3D representations that serve as features within the scenario, such as buildings, trees, signs, rocks or other 'props'. To add an object left click on Object in the Editor Object list, then double-click on the map or on the terrain in 3D. The Add Object dialog is very similar to the Add Unit dialog described earlier in this manual.

In the figure below is explained all the control in order to interact with the object:

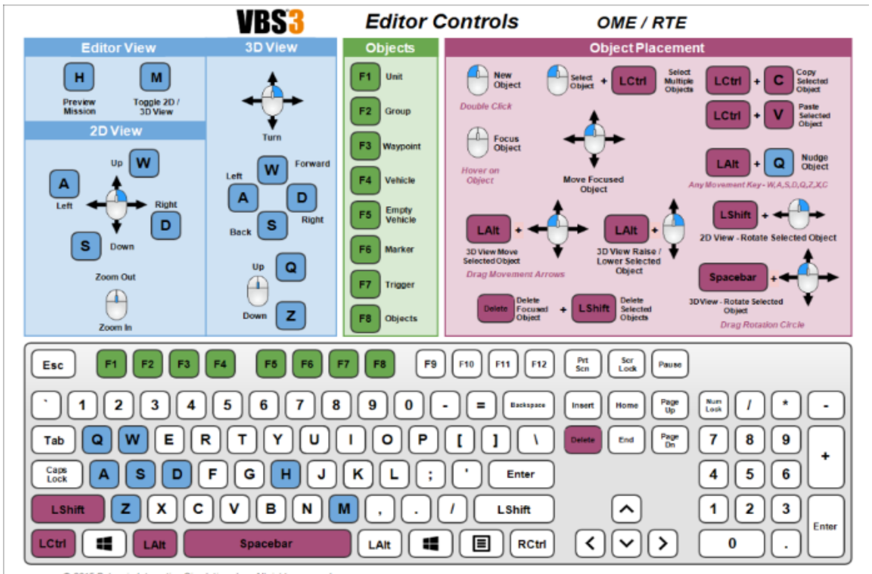


Figure 2.44 (VBS3 Manuals,2017)

2.4.1 Attaching objects

It is possible to attach objects, units and vehicles to other objects. Attached objects will maintain their offsets to the parent object when the parent object is moved.

To attach to an object, hold down <Shift> and left click on an object, vehicle or unit and then left click on another object.

For example, it is possible for objects with attachments to be attached to other objects, as shown in the Fig 2.46. The vehicle, fence and unit are attached to the tower and the tower is attached to the house. When the house is moved all other attached objects, units and vehicles will maintain their offset.

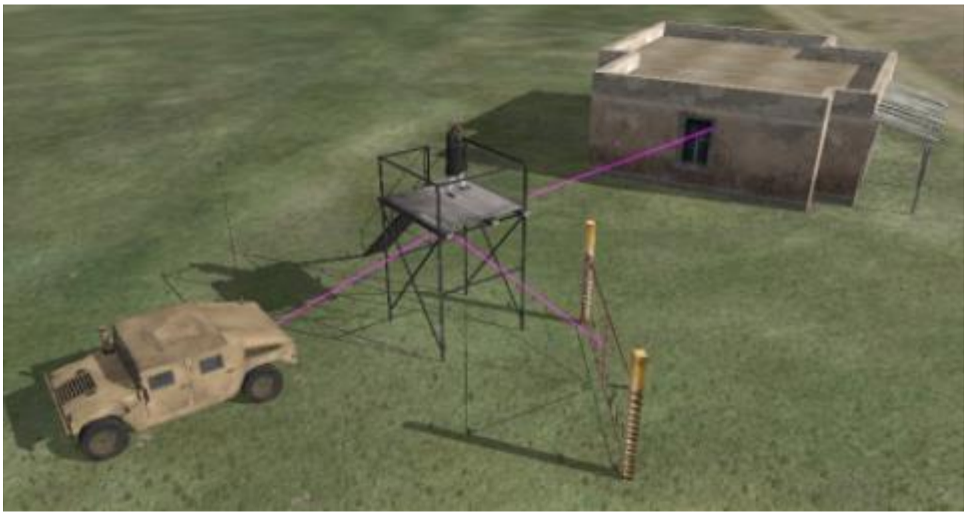


Figure 2.46 (VBS3 Manuals,2017)

In VBS there are a lot of options but it is not useful to explain everything.

In the following lines there are a list of the other options:

- Moves the camera 3D
- Moves the 2D
- Switch to Unit
- Snap to surface
- Above ground Off/ On

- Lock / Unlock object
- Deleting objects
- Scaling objects

2.4.2 Waypoint

Waypoints are used in VBS for units and vehicles.

The waypoints have to be attached to the units or vehicle. To add the waypoint in the 2D map: double-click with the right bottom of the mouse and select connect to vehicles or units.

The waypoints are used for playable and non-playable units/vehicles.

There are different types of waypoints, only the types that have been used in this work will be explained.

- Move: The unit/vehicle will move to this point or object. The move waypoint is considered complete when the unit gets close to the waypoint, the required distance being between 1 and 500 meters depending on the unit's vehicle type (if any) and whether a player is controlling the unit.
  If the waypoint is too far away or a valid path cannot be found, the unit/vehicle may never reach the waypoint. If a Move waypoint is attached to a building object, the "Position In-House" dialogue option will become available..
- Destroy
- Get in
- Seek and Destroy
- Join
- Join and land
- Get out: The group will move to the waypoint, then disembark from any vehicles its members are in. Helicopters will land on the closest "H pad" object within 500m of the waypoint. If any group members other than the leader are in a

vehicle of another group, that vehicle will stop to let them out. If the leader is riding in another group's vehicle, the Get Out waypoint will not be considered reached until the leader arrives at it under his own control (ie, never). In this case, a Transport Unload waypoint should be used. If the Get Out waypoint is placed on an object, the group will move to the location of that object at the instant the Get Out waypoint becomes its current waypoint, then disembark as normal.

- Cycle: This waypoint type will change the group's active waypoint to the nearest waypoint other than the group's previous waypoint. Note that the automatically created first waypoint (the leader's initial position as seen in the map editor) is considered as a Move type waypoint and can be used by the Cycle waypoint. A Cycle type waypoint can be used to make the group move in an infinite loop, a great and easy way to create a patrol. Either a Switch trigger or script can be used to "break" a group out of a cycling loop. In other words, The cycle waypoint creates a loop between the last waypoint (L) before the cycle waypoint (C), and the next closest waypoint (N) to the cycle waypoint (not counting waypoint L). The loop ignores any waypoints leading to the next closest waypoint (N) but includes all waypoints from the next closest waypoint (N) to the waypoint before the cycle waypoint (L).

  The red arrows in the following image show the loop that is created by the cycle waypoint (C):

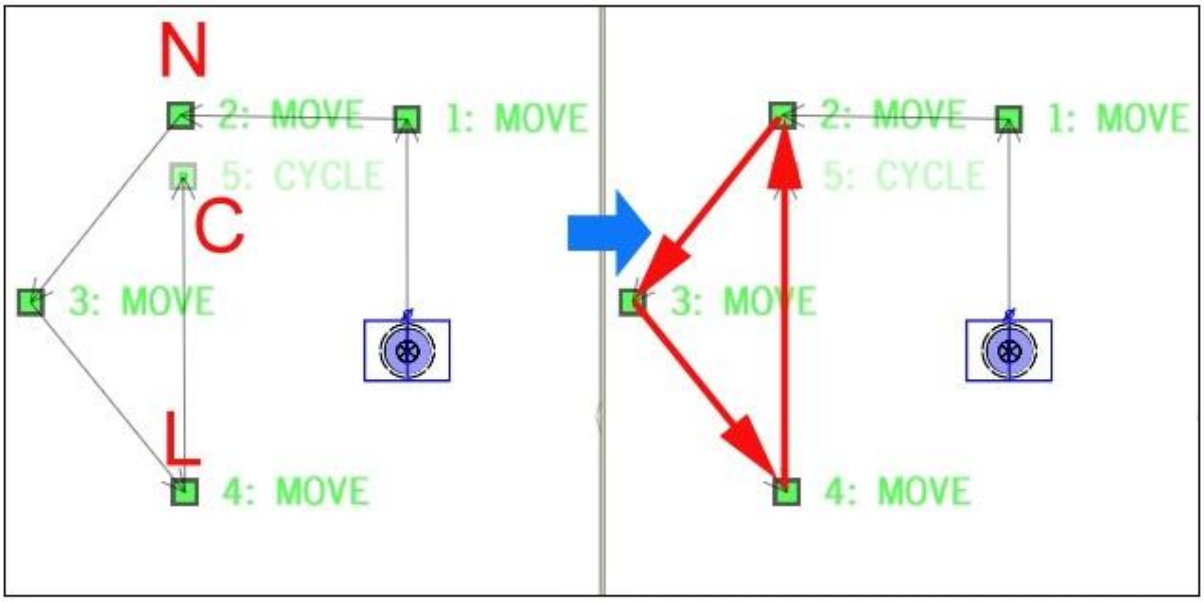


Figure 2.47 (VBS3 Manuals,2017)

2.4.3 Trigger

A trigger is an abstract game entity that waits until a certain condition returns true (for example, a unit or vehicle enters in a specific area) and then carries out a specific action (for example, once a unit or vehicle has entered a specific area, an enemy ambush commences).

The location and dimensions of a trigger are not always relevant, a trigger does not need to have any activation effects. Triggers may be linked to a unit or vehicle Editor Object, which means that only the linked unit or vehicle can activate the trigger.

In the Fig 2.48, the proprieties of the trigger are shown:

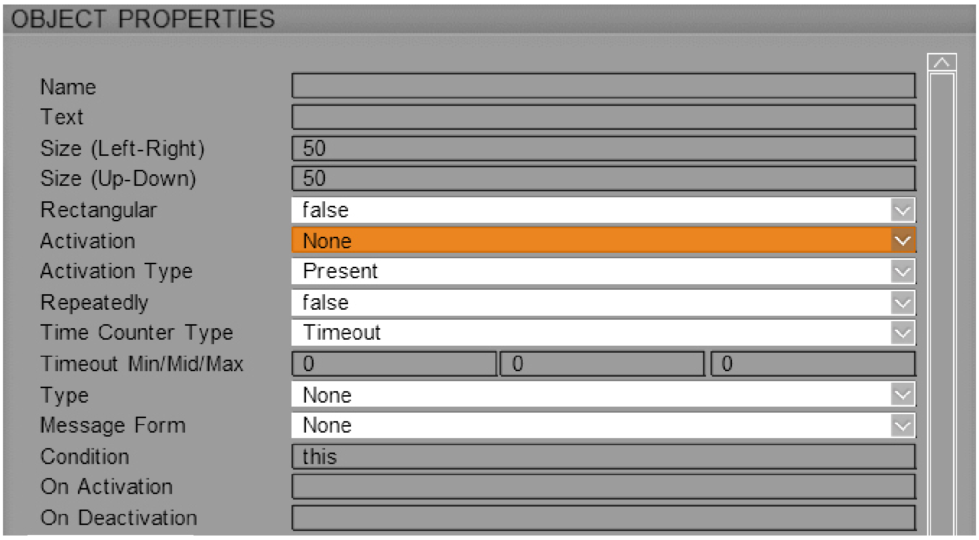


Figure 2.48 (VBS3 Manuals,2017)

Also, in this case, only the features used in the simulation will be explained.

- Name: Defines the name of the trigger, allowing it to be used in script code. The name must comply with normal variable name rules (no spaces, reserved characters or words allowed and duplicate name warnings may not give
- Text: This text is displayed in the OME / RTE when the mouse cursor hovers over the trigger, allowing the user to quickly identify what the trigger is and what it does.
- Size and Rectangular: The rectangular defines the shape and the sizes define the dimension
- Activation: Below, there is a list of activation method :
  1) Nome
  2) Opfor, Blufor, Civilian, Independent, Anybody
  3) Radio Alpha-Radio Juliet
  4) Seized by Blufor-Seized by Independent
- Activation type: If the trigger activator is a side or an object, the trigger is activated if that side/object is or is not present in the trigger area.
  To be considered present, an object must be alive (or not destroyed).
  The options are:
  1) Present
  2) Non Present
  3) Detected by BLUFOR - Detected by Civilians
  4) EngagementStart
  5) EngagementEnd

- Repeatedly
- Time Counter Type
- Timeout Min/Max/Mid
- Type
- Message from
- Condition: The trigger activates when this script code block returns true. In the script code block, the variable this refers to any conditions chosen in the activation options above, and thisList refers to an array of objects that are currently inside the trigger area and are on the activation side or option chosen (thisList does not always refer to units that are activating the trigger).

  If the activation type is not side related, thisList returns an empty array, and a seized-by-side trigger returns units of any side. If you leave the condition box blank, the trigger never activates. Using a script code based condition allows for the creation of more complex activation requirements, including multiple and/or conditions.
- On Activation: This script code block is executed when the trigger conditions are met, irrespective of the trigger type. Any actions defined by the triggers type take place immediately after this activation block begins.
- On Deactivation

In the following figure is shown how the trigger appears in the 2D view of the OME/RTE.

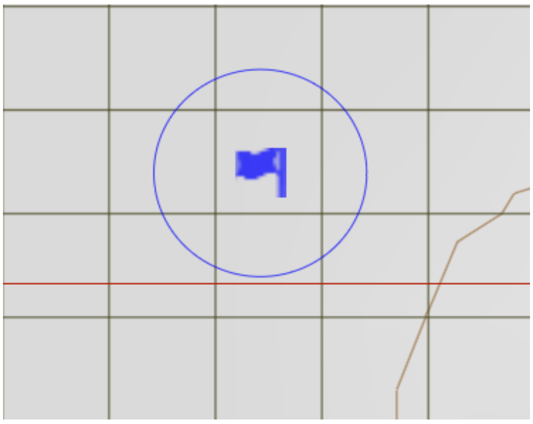


Figure 2.49 (VBS3 Manuals,2017)

2.5 Script

During mission editing, it is possible to come across situations where actions or features that can not be included using the Offline Mission Editor (OME).

The solution is to take advantage of the simulation engine's ability to use a more advanced feature known as scripting.

The VBS3 scripting language is called SQF syntax (where SQF stands for Status Quo Function). With SQF is possible to control and influence objects and the simulation environment with over 2,000 commands.

The ability to create objects and perform tasks not available in the OME allows trainers to enhance the training.

The simulation engine controls the environment and how objects will react in the environment. A scripting command is an instruction that tells the VBS3 simulation engine what to do. SQF is a proprietary language similar to C++.

A basic script can be added to any object in VBS, it is useful to get familiar with VBS script syntax in order to understand the operations made in this work.

2.5.1 Basic script syntax

The command and correct parameter must be used in the correct order.

A script statement can have three syntax variations, in order to script elements that will be recognized by VBS. It is presented in the following table:

SYNTAX VARIATION	DESCRIPTION
Command	The command takes no parameter. Example: `allUnits;`

Command
parameters
	The command takes one parameter, which can also be an Array of parameters
	Example:
	`Alive unit1;`
	`createDialog “mySampleDialog”;`

parameter1
Command
Parameter2
	The command acts as an operator between parameter and parameter2 (e.g. an arithmetic operator such as +, -, /, * is also considered a command), parameter1 and/or parameter2 can be also Arrays of parameters
	Example:
	`number1 + number2;`
	`unit allowDamage FAlSE;`
	`vic1 setDir 270;`
	`solider1 moveInCargo truck1;`
	`player say [“M11v02”, 5,  2];`
	`format [“Player:%1, player’s side:% 2”, player, side player];`

Table 2.1

Each script statement must finish with a semicolon (;) for the simulation engine to compile and execute it.

Any syntax variation has two constituents:

- Command: A specific VBS3 word and a command that does something in the simulation. Command names reflect what they do in the simulation (e.g. move a unit in a cargo position - moveInCargo, set a certain amount of fuel in a vehicle - setFuel).

  All commands are available in the specific document. This document is possible to find inside the VBS folder.
- Variables/Parameters (e.g. parameter1, parameter2,..): a single constituent needed for a scripting command to work. A variable can be a parameter for a command. Without the correct parameters, a command does not work. Each variable has its own data type. There are several data types that can be used with scripting commands.The following basic data types are introduced below
  1. Boolean: Can be a parameter for a command or a question to the simulation engine that returns TRUE or FALSE. Usually used in condition lines of triggers, waypoints or control structures.

     Command using data type Boolean:
     ```
     Unit1 allowDammage FALSE;
     ```
  2. Number: Any real number. (e.g. -1, 0, 1, 10.123)

     Command using data type number:
     ```
     Unit1 setDir 270;
     ```
  3. String: Specific text enclosed by "quotes" or 'apostrophes'. Usually used with messages displayed on the screen. When scripts use markers, the "marker name" is a string. Names from the .pdf references (weapon, magazine or vehicles) are strings when used with scripting

     Example: "marker_name", " VBS3_AKM", "my message to be displayed".

     Command using data type String:
     ```
     Unit1 addWeapon ‟ VBS3_AKM”;
     ```

4. Object: Name property of any object (unit, vehicle, trigger, building, etc) added in the editor.

   Example: unit1, truck1, bldg1

   Command using data type object:

   ```
   Unit1 moveInCargo truck1;
   ```

5. Side: The side an object belongs to. Most common are EAST, WEST, CIVILIAN, these are not text strings.

   Command using data type Side:

   ```
   Temp_grp = createGroup WEST;
   ```

6. Group: Defined in the simulation as a collection of several units or a single named unit, in the following figure is shown an example of group names in VBS3.

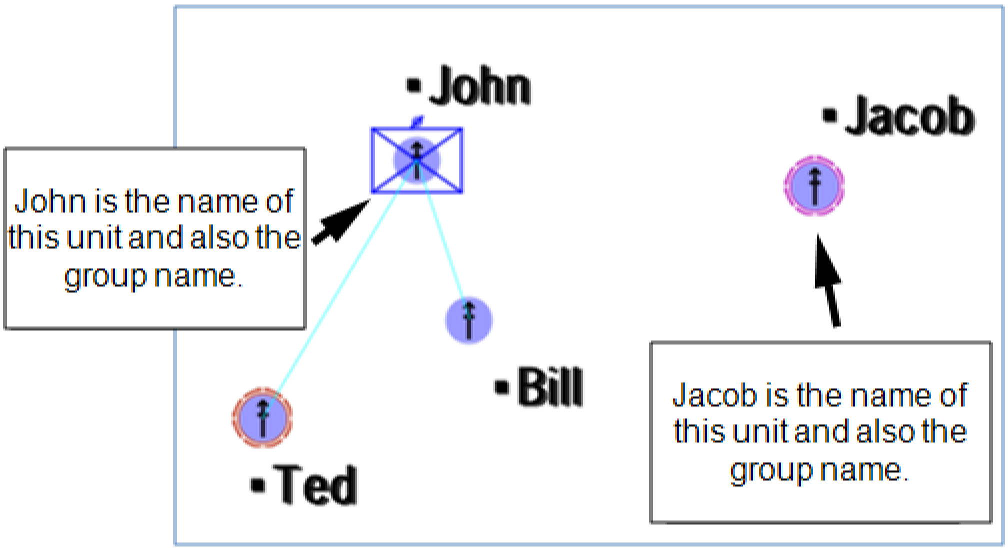


Figure 2.50 (VBS3 Manuals,2017)

7. Array: List of items enclosed by square [brackets]. Each element is separated by a comma. Usually a list of items such as names, strings, or a position.

Example: [unit1, unit2, unit3] or [" VBS3_AU_AW50", " VBS3_AKM"] or [[" VBS3_AU_AW50", 3], [" VBS3_AKM", 2]]

Command using data type Array:

```
Unit1 setPos [10, 10, 0];
```

In any syntax variation, there are two general rules about using spaces, these rules are explained in the following table:

RULES	EXAMPLES
Spaces are needed before and after scripting commands	unit1 moveInGunner vic1; `// correct` unit1moveInGunner vic1; `// incorrect, need a space before the command` player sideChat "working"; `// correct` playersideChat "working"; `// incorrect, using two commands, need a space in between each one` vic1 setDir 270; `// correct` vic1 setDir270; `// incorrect, need a space after the command`

Spaces are optional before and after special characters
● Math: + - * / = < >
● Punctuation: , ; ""
● Brackets: [ ] { } ()
	`var1 = direction vic1*speed vic1;`
	
	`this addMagazine "ak47_mag";`
	
	`if (player in vic1) then {}`

Table 2.2

2.5.2Basic variable parameter

Variable stores a value. The value can be a number, text, or data of a more complex type. When a variable is used with a script command, it is also a command parameter. Unlike in programming languages such as C++ or Java, the type of the variable does not need to be explicitly stated when the variable is declared. Instead, the type is automatically inferred from the assigned value.

In contrast to languages such as JavaScript or PHP, there is no implicit type conversion. This means that if a script command expects a String and is given a Number instead, it does not automatically convert the number to a string but rather generate an error.

Variables are automatically declared when assigned to for the first time.

However, there are some rules for variables name declaration:

- Only text characters and numbers (a-z, A-Z, 0-9).
- No special characters (_).
- Cannot start with a number or underscore
- No spaces
- No scripting commands and no duplicates.

In order to achieve a variable the operator = it has to use. For example: `myVar = 3.`

It is useful to assign a variable like typeName. In this way, it is more difficult to make an error.

2.5.3 Local and Global Variables

There are typically two types of variables: Global and Local.

Global variables can be accessed from any scope. For example, if declared within a script file (for example, init.sqf), a global variable will be visible to code in any script. This makes global variables very dangerous to use because you can cause a name collision if there are two instances of the same script running and both scripts are trying to use the same variables. Because of this, global variables should be avoided wherever possible.

Local variables exist only within the scope where they are declared. For example, if declared within a script file (for example, init.sqf), local variables will only be visible to code within that script.

Local variables are declared by prefixing them with an underscore (_).

For example:

```
myGlobalVar = 10; // This global variable persists outside of this scope
_myLocalVar = 3; // This local variable can only be accessed within this script. When this script finishes executing, the variable will no longer exist
```

The last types of variables are explained below (special variables in VBS)

This, _this and thislist:

- this: The keyword is used in the initialization of objects, and refers to the object itself. It is also used in Waypoints and Triggers and designates the Waypoint or Trigger objective, respectively.
- _this: The keyword is used for passing arguments to other scripts.
- thislist: The keyword is used in Waypoints and Triggers. In Triggers, it refers to all the objects which fulfill the condition of a Trigger. In Waypoints, it refers to all the objects that have completed a Waypoint.

2.5.4 Operator

Operators are the basic commands for each programming language, including SQF, is built on. They provide the ability to perform basic mathematical and logical operations.

Terms:

- Operand: An operation is any value given to an operator.
- Expression: An expression is any code that returns a value.

- Nullary Operator: Just the operator, without any operands, which makes it any engine-defined variable.
- Unary Operator: A unary operator is an operator that requires only one operand.
- Binary Operator: A binary operator is an operator that requires two operations.

In order to assign an operator the operator (=) has to use. In other words, the general syntax is:

variable = expression

There are the following types of the operator:

- Arithmetic operators: All operator must be numbered and always return a number
- Logical operator: Logical operators evaluate Boolean values. All operands of logical operations are Booleans. A logical operation always returns a Boolean.
- Array operators: SQF offer its own operators to handle arrays. All operators have to be of type Array. The return value of an array operation is an Array.

All script have to save in VBS folder . After VBS will interact directly with the script. The first file that has to create is the Init.sqf

2.5.5 Execution

The Init.sqf file allows the execution of script statements before the mission starts (i.e. when the mission is initialized). It is an alternative to the Initialization Statement field. If this file exists in the parent mission folder, when the scenario starts, all computers connected to the server will execute the Init.sqf file.

In other words, if there is a script with whatever name (the name of the file.sqf), this name has to be present in the Init.sqf, exactly with the following syntax:

`Namefile`.sqf

In the lines before the initialization statement has been mentioned, it is a field of every object is the most common place to insert a basic script. Any script inserted in that field executes before the mission starts. The field can use 'this', like in the figure below.

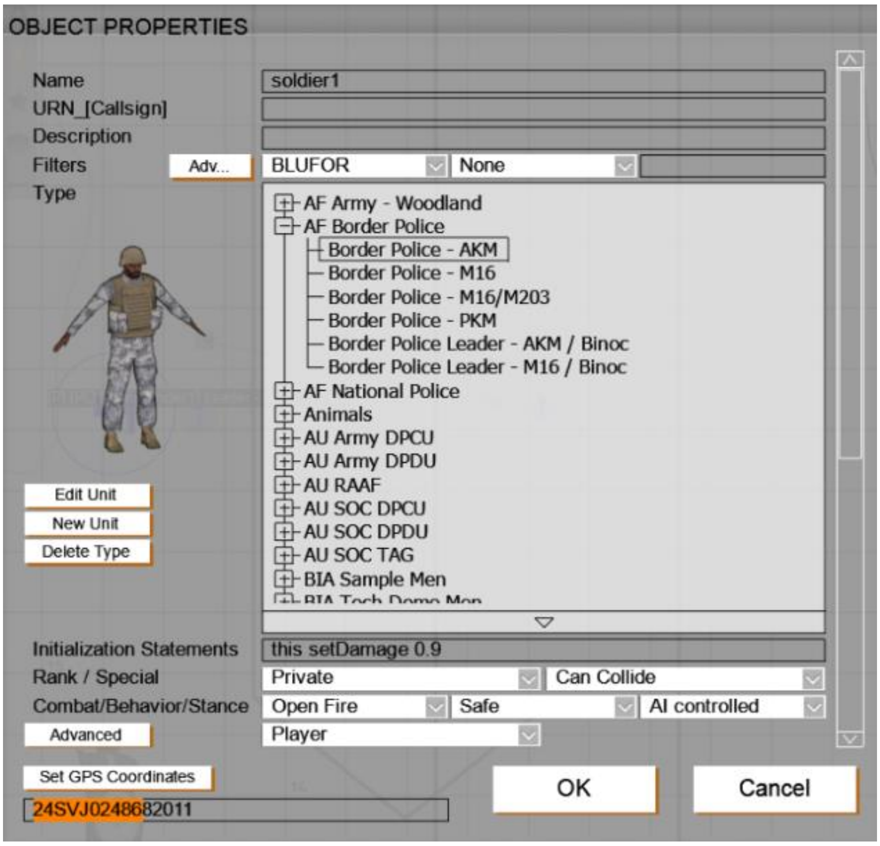


Figure 2.51

2.6 Training Scenarios

In order to train and show the helicopter simulator three different scenarios have been created: one for multiplayer and two for singleplayer, working with different logics.

The map in which the scenarios have been implemented is common for the three scenarios. The map's name in VBS3 is EasternEuropeanGeotypical25km (Fig 1.42).

Figure 2.13

2.6.1 Multiplayer Scenario

The scenario has been designed to be used by two trainees. The trainees will work on different computers connected by a local network.

The players can choose which will be the pilot and which will sit at the crew station.

Evidently the performance of the one will influence the performance of the other player.

In the first part of the simulation the pilot will have to follow a ghost helicopter in front of it until it arrives to the landing point. The second player will wait at the landing point and will be able to enter inside the helicopter once it has landed. The ghost helicopter will stay there and the pilot will have to drive the helicopter to accomplish several targets.

The targets created inside the test field are of different nature. Both static and moving targets have been implemented.

Once all the targets are completed the pilot will have to return to the airport.

<u>Logic Implementation</u>

Below is shown how the whole map looks like once the logic has been implemented.

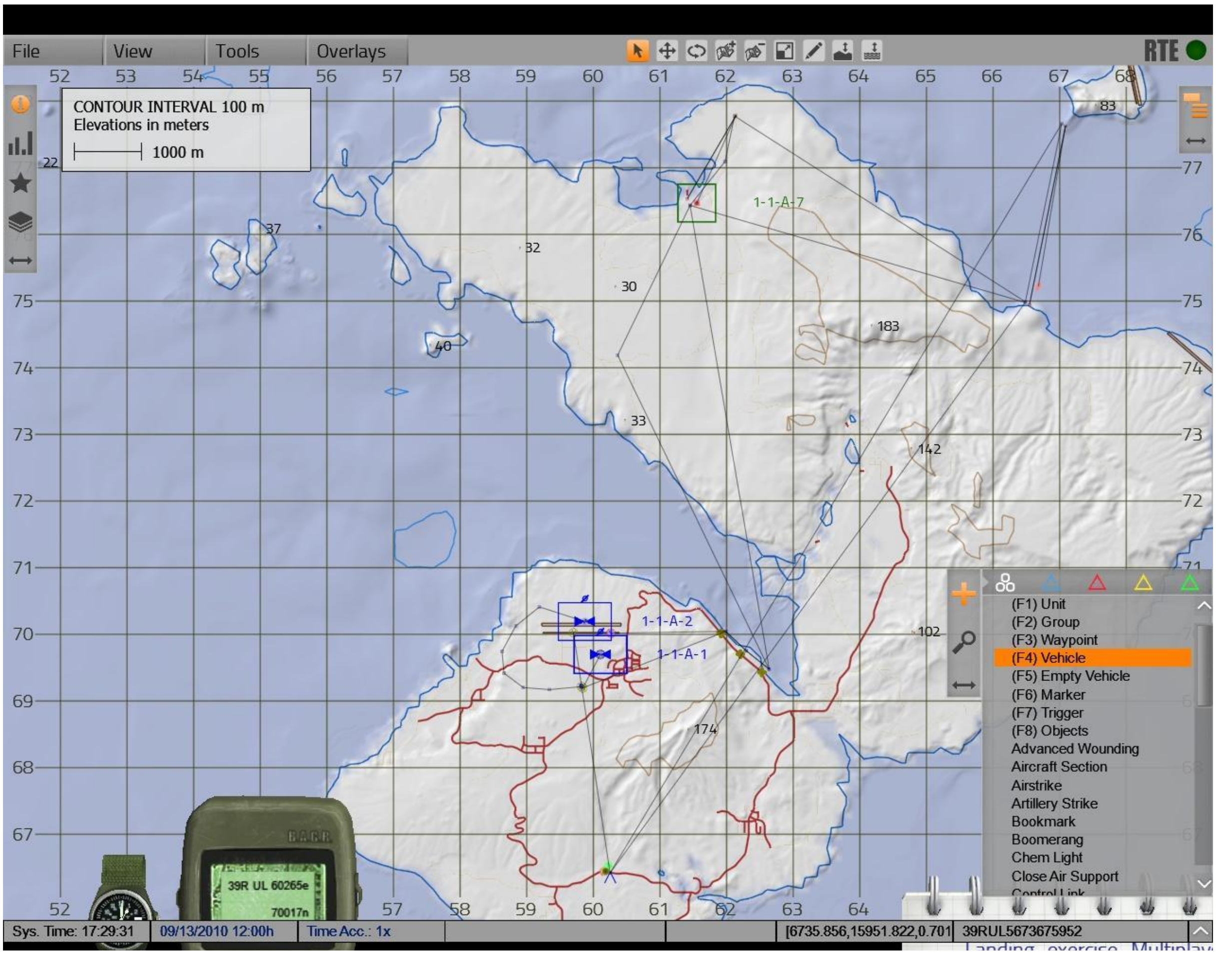


Figure 2.14

It will be explained in detail how the creation has been done indicating the explained quarter using the referenced squares in the map.

<u>Flying and Landing Exercise, Fig. 1.43 quarter (58-61;69-71)</u>

The ghost helicopter called in the map 1-1-A-2 is driven by AI. The implementation of the automatic pilot has been done using two type of Waypoints: Move and Land. It will appear to the trainee, that chose to be the pilot a message, hinting him to follow the helicopter and in case of fail to directly go to the landing point.

In order to make the helicopter follow a determined path the Waypoints can be placed in the desired locations and then they can be connected one to the other simply clicking on one with the right button of the mouse and choosing the option called Add Existing Waypoint. Once the Waypoint are all connected the will be enumerated in the desired sequence.

The landing point for the trainee is implemented too with a Waypoint Land that will show in the headset the distance and the direction for the pilot.

Once the pilot is near the landing point a big H where the helicopter can land can be seen on the ground.

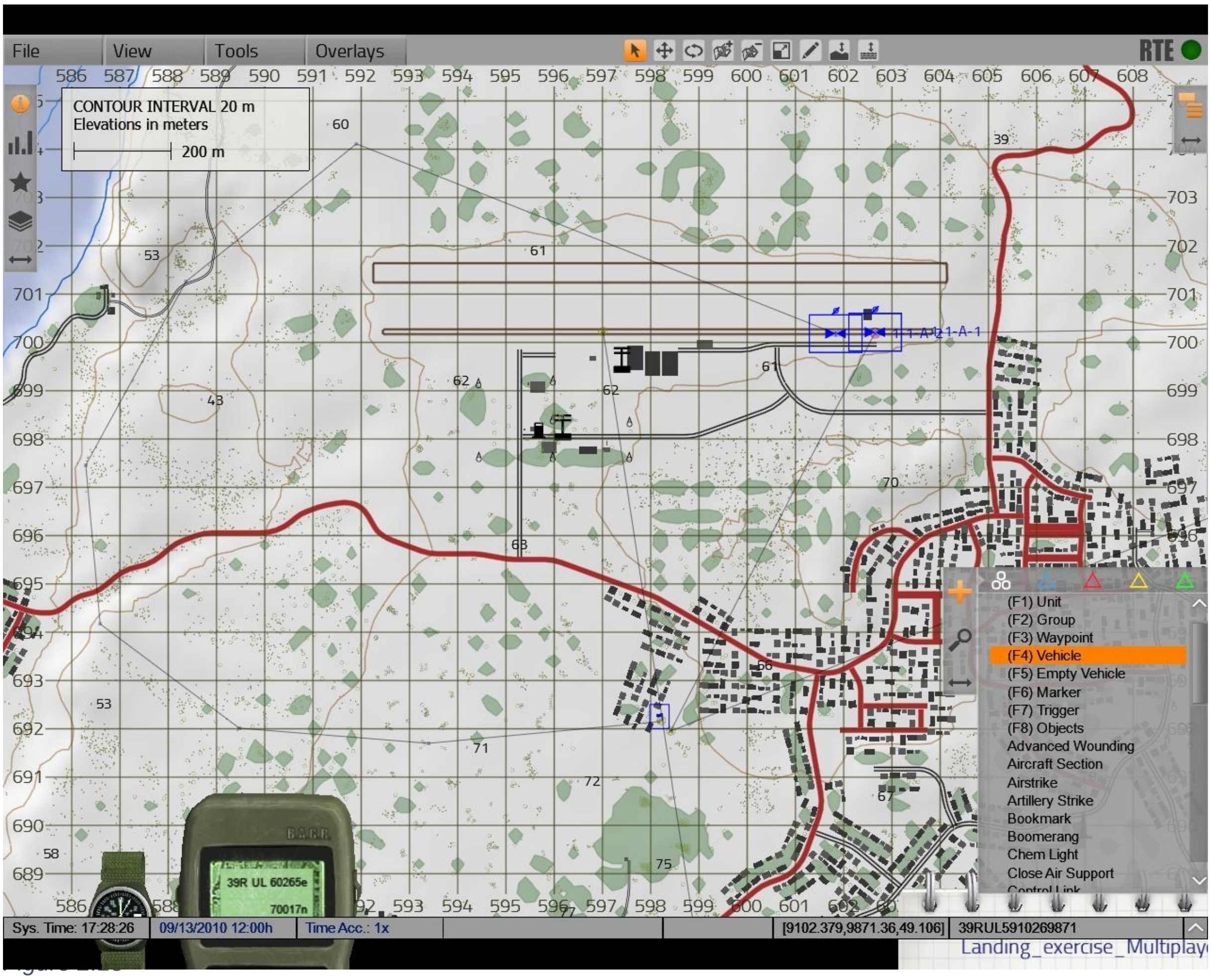
The second will be placed there and will be able to enter in the helicopter.

In Fig (1.43) looking at the quarter (598-599; 692-693) can be seen a trigger. This trigger was used to implement two actions for the ghost helicopter: set the position of the ghost helicopter in the landing area and to get off the aircrew. The implementation has been done writing the following code in the On Activation field inside the editing window associated to the trigger:

```
doGetOut AutoHeli; AutoHeli setPos pos;
```

Explainig the syntax:

- doGetOut is the command to get off the aircrew,
- setPos is the command to set the position of helicopter,
- Pos is a command that get the position of the trigger
- AutoHeli is the name that has been assigned to the helicopter.



Static targets, Fig. 1.43 quarter (61-63;69-71)

After landing and getting in the second player the pilot will have to fly to the first's targets. The path to be followed will be shown to the pilot using three Waypoint Move. The Waypoint are set in front of three sets of barrels that are the static targets for the second player.

The waypoint can be seen in Fig 1.45 as the blue rectangles linked one to the other by one line, the barrels are represented by the green points placed on the red street along the coast.

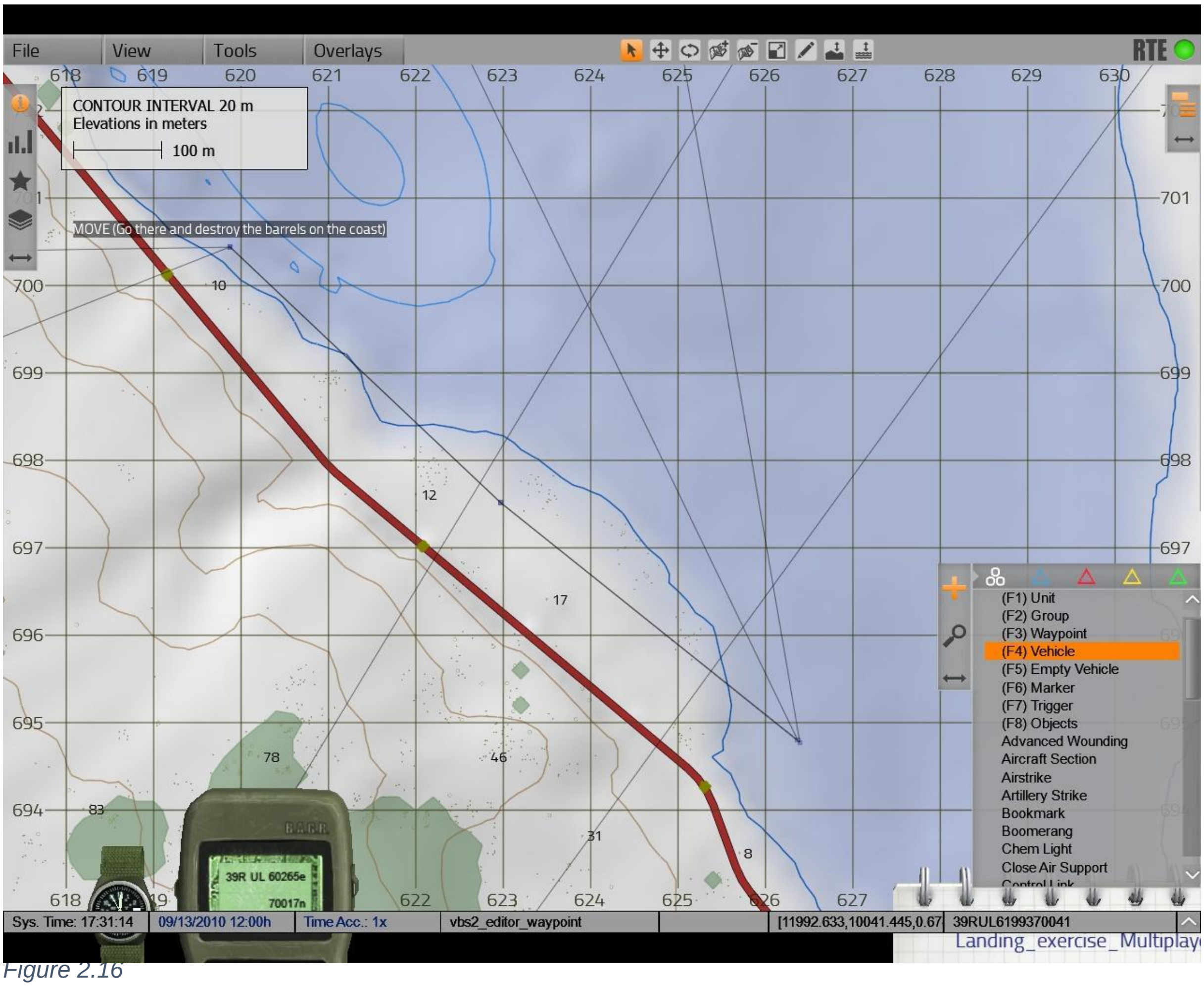


Figure 2.16

In Fig 1.46 can be seen how the barrels look like during the simulation.

Figure 2.17

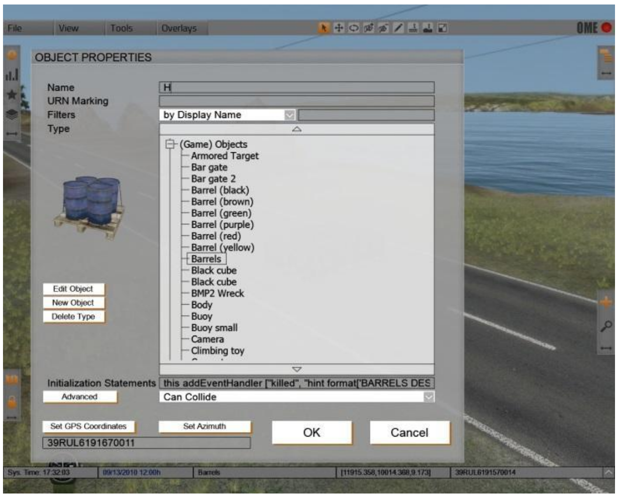


Figure 2.18

The players are advised when the targets are destroyed by a message that will appear on the screen. This feature has been implemented by adding an event handler to the barrels. The implementation of the event handler has been done writing the following code in the Inizialization Statement properties window associated to the trigger with the following syntax:

this addEventHandler[“killed”, “hint format[BARRELS DESTROYED]”];

Explaining the syntax:

- ´this´ is a keyword in VBS3 that is used to indicate to the computer to refer to the object
- addEventHandler is simply used to add the even handler
- “killed” is a type of event handler that activates when the object is associated at is destroyed
- “hint format[]” is a command to print a message on the screen

Moving targets controlled by waypoint, Fig 1.43 quarter (61-63;76-78) and (66-68;75-78)

In order to increase the difficulty two moving targets have been implemented. Two different types of moving targets are in the simulation: one moving boat and one car.

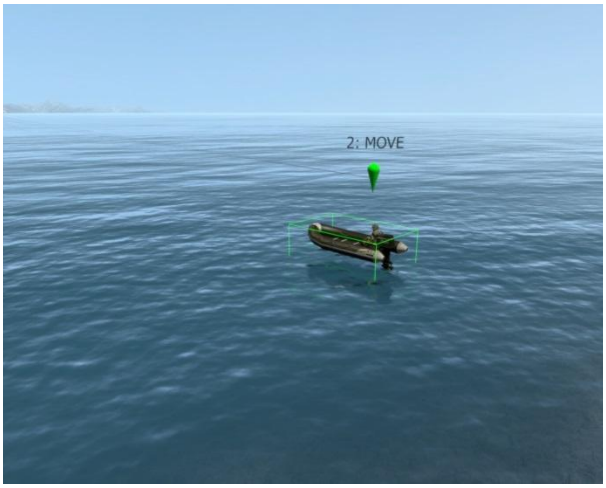


Figure 2.19

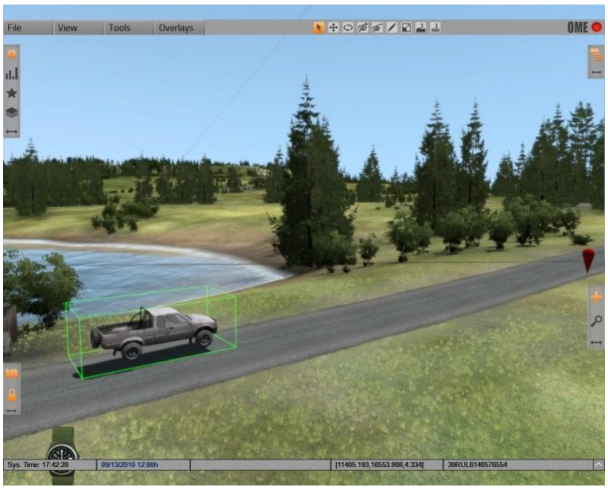


Figure 2.20

Both of them are automatically guided by AI. In order to make them follow a fixed path some Waypoint Move linked to one Waypoint Cycle have been implemented. In this way all the vehicles will cyclically move from one point to the other.

Figure 2.21 – Terrain Path Visualization

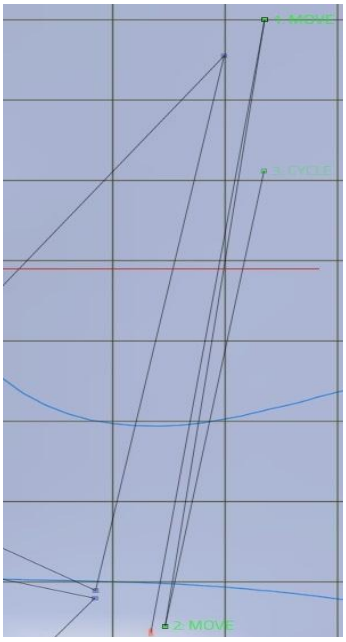


Figure 2.22 – Sea Path Visualization

Moving targets controlled by function, Fig 1.43 quarter (60-62;66-68)

In the last part of the training mission are implemented 6 moving targets. They can be identified in Fig 1.52 by the red rectangles. The difference from the previous moving targets is that these are controlled by an external function and not by Waypoints. Indeed the units are controlled by a patrol function called FunctiOnLoad.sqf in the mission folder, while the function that actually move the soldiers is called Array.sqf. The entire scripts can be found in Appendix 1 and Appendix 2.

The units will turn around an ammunition box never exceeding the desired distance set in the scripts.

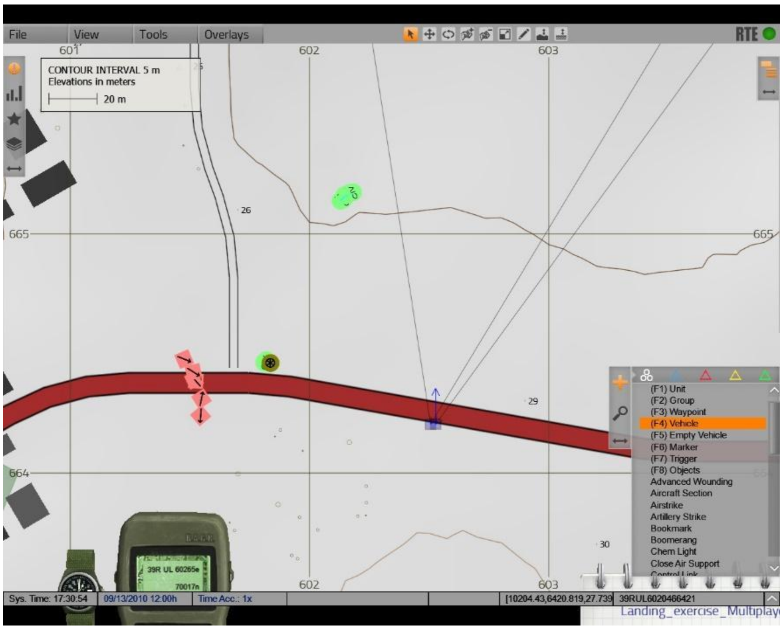


Figure 2.22 – Mission View VBS3

2.6.2 Singleplayer Scenario with Syncronized Waypoints and Triggers

This scenario has been designed to be used by one player at the gunner position. The helicopter will be driven by the automatic pilot.

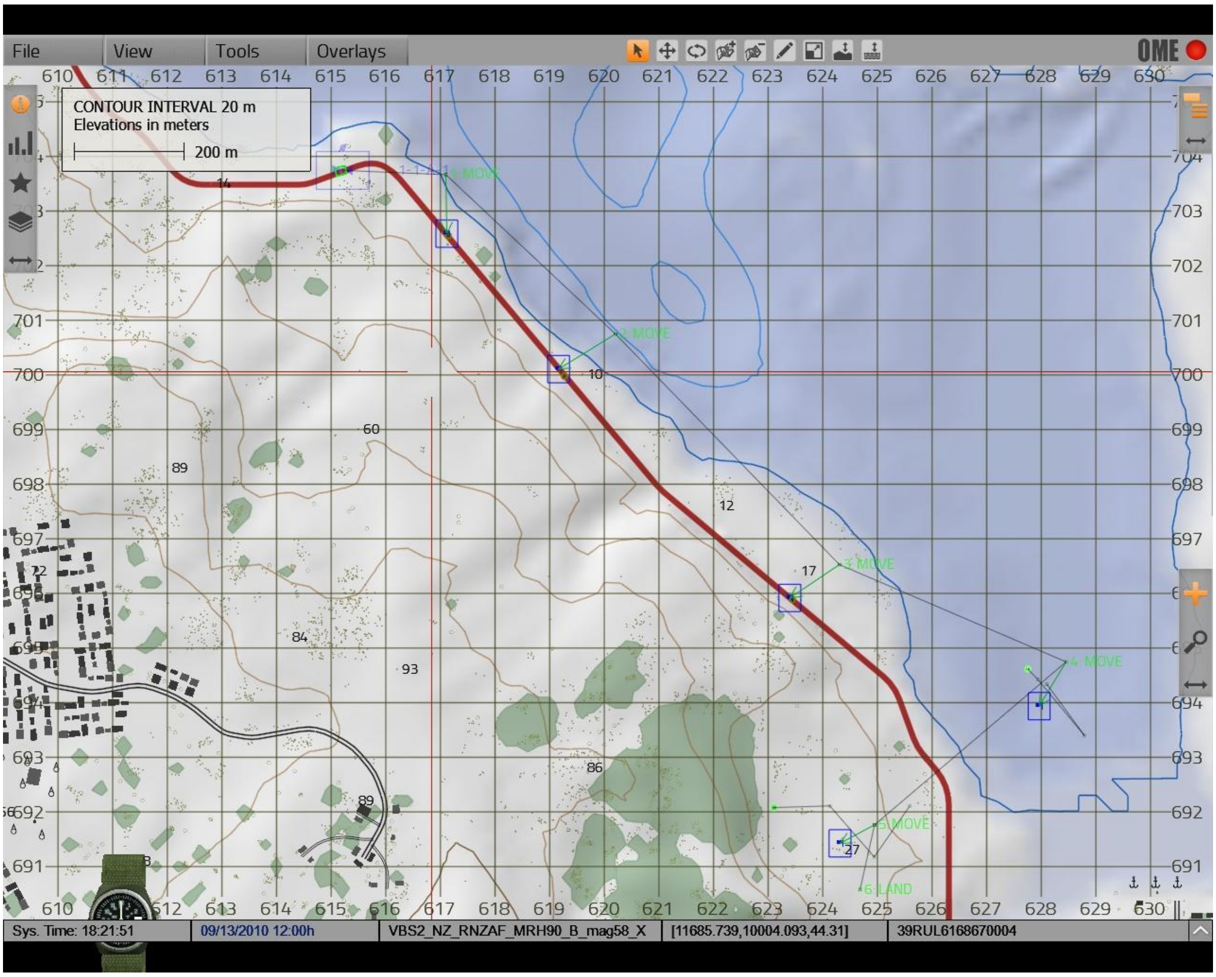


Figure 2.23

The helicopter will be driven by the Wapoints Move that in Fig 2.23 are represented with green numbers and text. The trainee will have to complete all the triggers that are synchronized with the Waypoints.

The aircraft heigh has been set in the Waypoints object properties window at a value equal to 30 [m]. This has to be done because the default heigh is 150 [m] and it was too high to accomplish the triggers.

Syncronization example, Fig 1.53 quarter(615-618; 702-704)

An example of Syncronization between Waypoint and Triggers can be seen in Fig (2.24). It works like this: the simulation will start with the helicopter taking off and moving to the first Waypoint called 1 MOVE. As the Waypoint and the Trigger are synchronized the helicopter will stop where the waypoint is placed until the Trigger is activated. If they are not synchronized the helicopter would just go straight to the next waypoint.

The synchronization can be visually seen in Fig(2.24), it is represented by the green arrow that connects the Waypoint to the Trigger.

Figure 2.24 - Synchronization

In order to activate the Trigger and let the helicopter go to the next waypoint and the next trigger, it is necessary to destroy the car that is parked along the coast. The

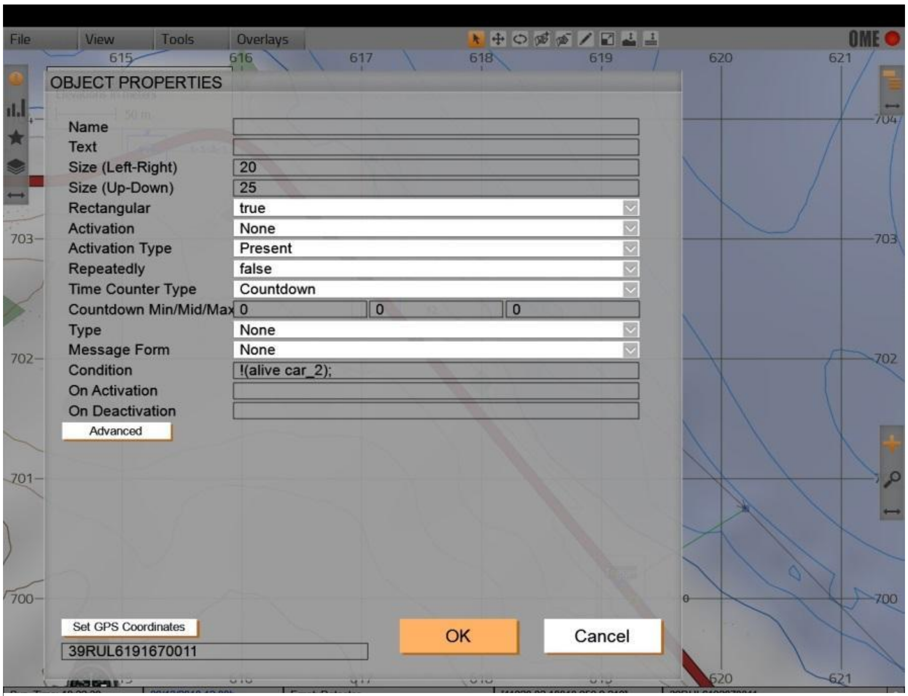


Figure 2.25

implementation of it can be seen in Fig 2.25.

The Trigger activates when the code written in Condition field inside the object properties of the Trigger is true. In this case the code is !(alive car_2). So it activates once the car is destroyed.

<u>Moving targets, Fig 1.53 quarters (623-626;691-693) and (627-629;693-695)</u>

Two moving tagets has been implemented again: one boat and one truck. How the implementation looks like in the editor is shown in Fig 2.26 and Fig 2.27.

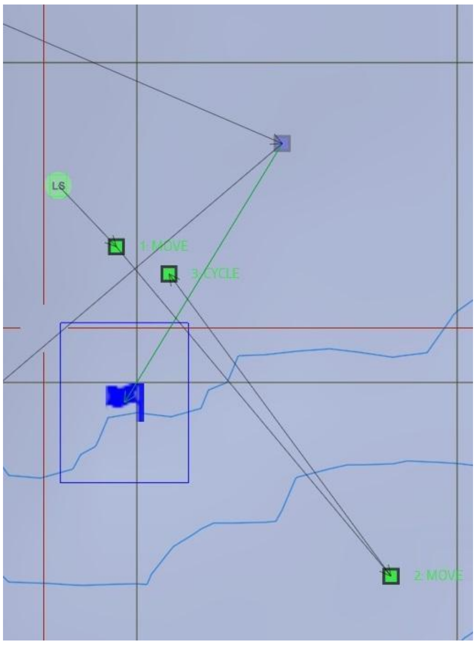


Figure 2.26

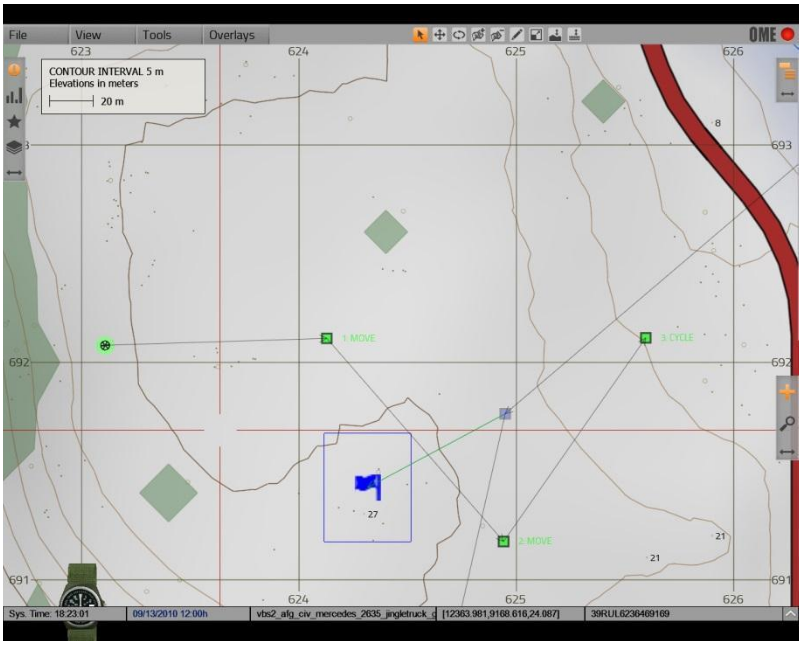


Figure 2.27

The logical implementation is the same for both vehicles:

- The blue rectangle with the grey arrows connecting it is the one of the Waypoint Move linked to helicopter
- The flag represent a Trigger that is synchronized with the waypoint linked to helicopter and the synchronization works exactly like explained above
- The green circles present in both figures represent the vehicles
- The green rectangles linked by arrows each other and to the vehicles represent the Waypoints liked to the targeted vehicles, the scheme used is the one about the Waypoint cycle: the truck in Fig. 1.57. will move along the triangle shaped by the Waypoint 1 Move, 2 Move and 3 Cycle. The boat in Fig. 1.56 will go backward and forward again between Waypoint 1 Move, 2 Move and 3 Cycle.

2.6.3 Singleplayer Scenario using Triggers

The single player scenario is designed only for the gunner player. In the helicopter, there are two gunners, one on the left side and one the right size. This scenario is made in order to play on the right side. The helicopter is autopiloted and it will move to the next task only when the previous is done. (It will be fully explained afterward).

The targets created inside the test field are of different nature. Both static and moving targets have been implemented.

As said before, this simulation is made for the training of the soldier at the crew station, each target has been made in order of difficulty. A list of the different targets that the trainee will find during the simulation is reported below:

- Three cars (static) in the middle of the road.
- One truck moving along the road
- One boat moving between two coasts.
- One barrel in the middle of the road.

In order to increase the difficulty of the different targets three parameters have been taken into account:

- Dimension: The dimension of the different objects is in descending order. In the beginning, the trainer can destroy the car very easily.
- Static and moving: Some targets are static and other moving.
- The distance between the helicopter and the target: As said before, the helicopter is autopiloted. The distance between the targets and the helicopter is controlled by external scripts. In the beginning of the mission, the helicopter will be closer to the targets then in the next ones.

Logic Implementation

In order to insert commands and messages for the player an external script has been used.

Inside the game there are Triggers and one Waypoint, the Triggers are connected with the script by calling them with the name assigned in VBS3´s editor.

Below the script is presented. In the next pages, each part of the game will be explained referring to the line number of the script, in order to understand how the script affects the simulation.

```
1  hint "target 3 cars in the middle of the road";
2   _counter = 0;
3
4  waitUntil { !(alive car_1) };
5  _counter = _counter + 1 ;
6  NH90 doMove getPos point_2;
7  hint format ["Target eliminated: %1/6",_counter];
8  sleep 2;
9  hint "next car";
10
11 waitUntil { !(alive car_2) };
12 _counter = _counter + 1 ;
13 NH90 doMove getPos point_3;
14 hint format ["Target eliminated: %1/6",_counter];
15 sleep 2;
16 hint "next car";
17
18 waitUntil { !(alive car_3) };
19 _counter = _counter + 1 ;
20 NH90 doMove getPos point_4;
21 hint format ["Target eliminated: %1/6",_counter];
22 sleep 2;
23 hint "this task is completed!Destroy the moving car ";
25 sleep 15;
26 hint "The car is moving along the road, destroy it";
27
```

```
waitUntil { (damage car_4)>0.1 };
_counter = _counter + 1 ;
NH90 doMove getPos point_5;
hint format ["Target eliminated: %1/6",_counter];
sleep 2;
hint "this task is completed!Destroy the moving boat ";
sleep 15;
hint "The boat is moving, destroy it";
waitUntil { !(alive boat_1) };
_counter = _counter + 1 ;
NH90 doMove getPos point_6;
hint format ["Target eliminated: %1/6",_counter];
sleep 2;
hint "this task is completed!Destroy the barrel in the road! ";
sleep 5;
hint "There is a barrel on the road, search it";
waitUntil { !(alive barrel)};
_counter = _counter + 1 ;
NH90 doMove getPos point_7;
hint format ["Target eliminated: %1/6",_counter];
sleep 2;
hint "Your mission is done! Wait that the helicopter come back to the  airport";
```

Below is shown how the whole map looks like once the logic has been implemented.

Figure 2.28

<u>Start mission, Fig 2.28 quarter (60-61;60-71), script lines (1-2)</u>

It is the start of the mission, the message “destroy the 3 cars in the middle of the road” will appear on the screen, thanks to the line 1 in the script. The command used is hint. hint is a command in VBS that allows to print a message in the screen.

In line 2 of the script there is a variable initialized at zero (`_counter = 0;`), this variable will be necessary for the completed target´s counter.

The NH90 will start moving thanks to the Waypoint, it is the only action in the simulation that does not come from the script. Why the Waypoint has been used will be explained later.

<u>Static targets: cars, Fig 2.28 quarter (61-63; 69-71), script lines (4-27)</u>

In the beginning, the helicopter will reach the Waypoint. The Waypoint was implemented in order to control the Aircraft Height. It was not possible to do it only using Triggers.

In order to set the flight height, the option "Aircraft Height" must be edited, it permits to set the flight height in feet.

The options presented in the object properties window relative to the Waypoint are shown in the figure below.

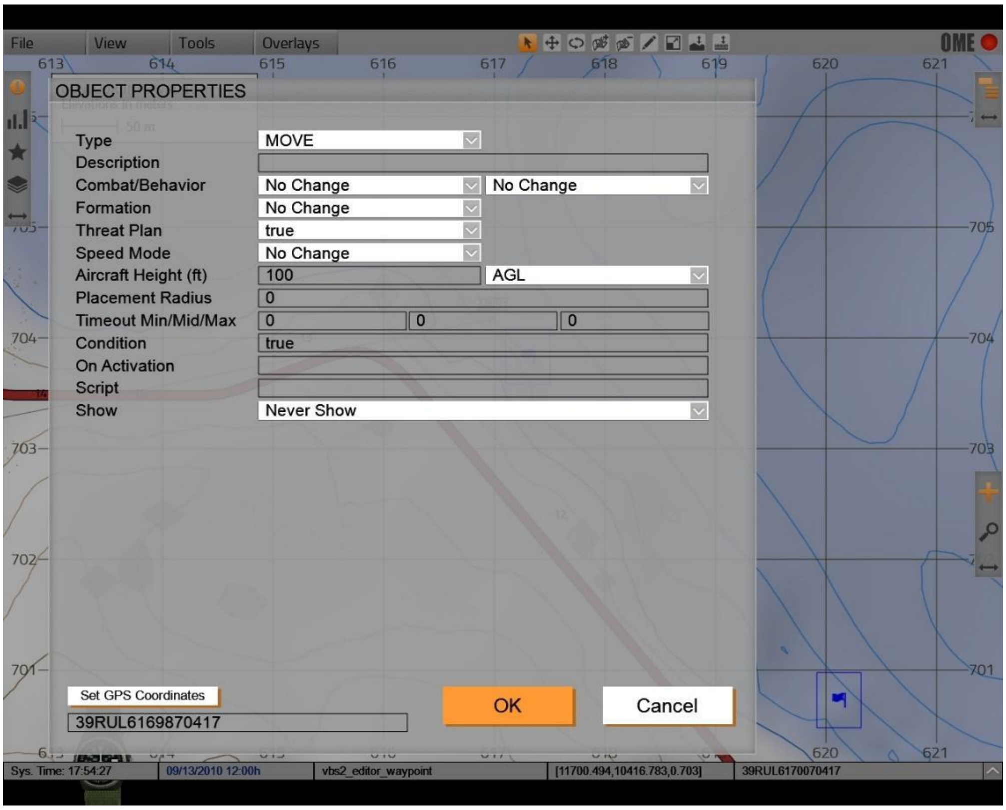


Figure 2.29

In Fig. 2.30, is shown how the waypoint will look like in the 2D map of VBS3 editor.

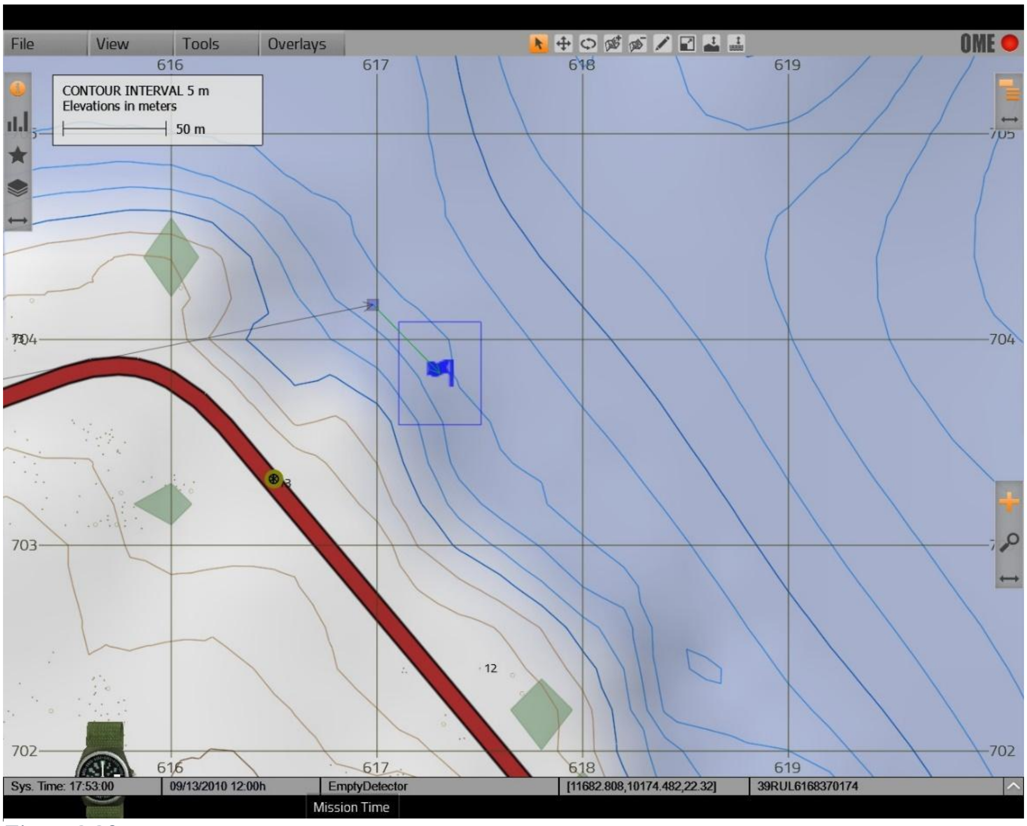


Figure 2.30

The rectangle with the flag is the Trigger and the little square with an arrow attached is the Waypoint. After this point, thanks to the waypoint setting, the average high of the helicopter will 30 [m].

In the line 4 of the script, there is the command **`waitUntil { !(alive car_1) },`** thanks to this command the helicopter will remain stationary until the car is destroyed.

The command **`NH90 doMove getPos point_2;`** visible in line 6 of the script, is the basic command for moving the helicopter. It is better to explain it in full.

- NH90: it is the name assigned to the helicopter in VBS3 editor
- doMove: it is the command for moving the helicopter to a specific point.

- getPos point_2: it is the command for getting the position of the specific point, in this case the position of the Trigger and its name in the editor is point_2. Therefore, the Trigger is useful for getting the position from the editor and pass the position to the script.

In summary, the basic controls for the helicopter are whaitUntil, doMove, getPos. These commands will be repeated inside the code every time the helicopter has to move to another position. Obviously, they will not be explained again.
In line 5, there is the variable previously mentioned (_counter) , and it is incremented by one. This variable will be displayed on the screen, thanks to the command in line 7 (**`hint format ["Target: %1/6",_counter]).`**
So, in the screen will be printed: Target eliminated: 1/6. This is how the counter works, the trainee can understand how many targets are completed about the total (6).
The line 8, there is the command sleep, it is used in order to make the messages appear in sequence.
In the specific (**`sleep 2;`**), the number 2 is the time in second. So in this case, line 9 will be executed after 2 seconds.
Until the line 26, the code will be the same, with an orderly repetition of commands.
There are three cars in total, for each car there is a trigger for the helicopter. Like in the figure below:

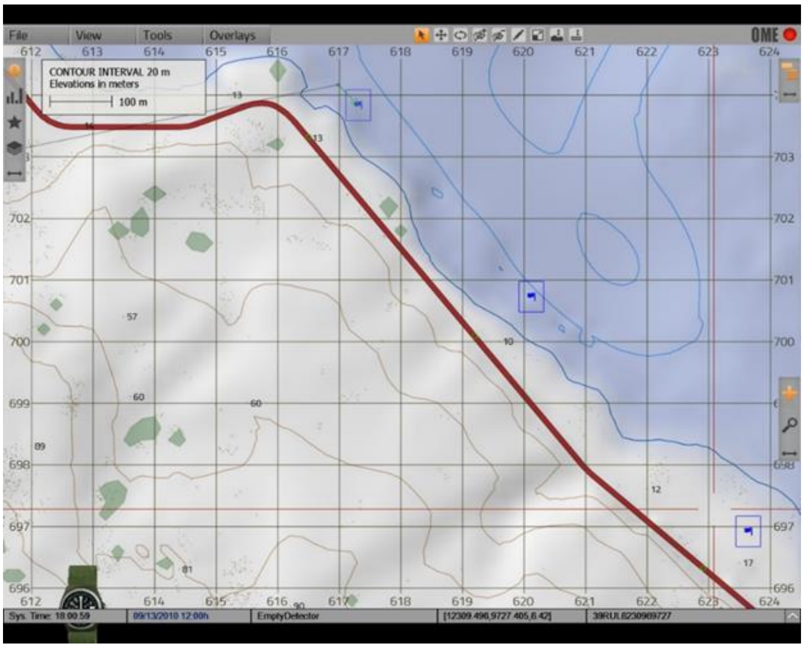


Figure 2.30

A green point in the maps represents the cars. In the game the cars are like in Fig. 2.31:

Figure 2.31

Moving target: truck, Fig. 2.28 quarter (63-65; (69-71), script lines (28-35)

In this part of the game, there is a truck that will be moving back and forth along a defined path.

In order to move a truck three Waypoint with cycling option have been used.

In the two figures below is shown, how the truck looks like during the simulation and how the truck linked to the Waypoints looks like in the editor.

Figure 2.32

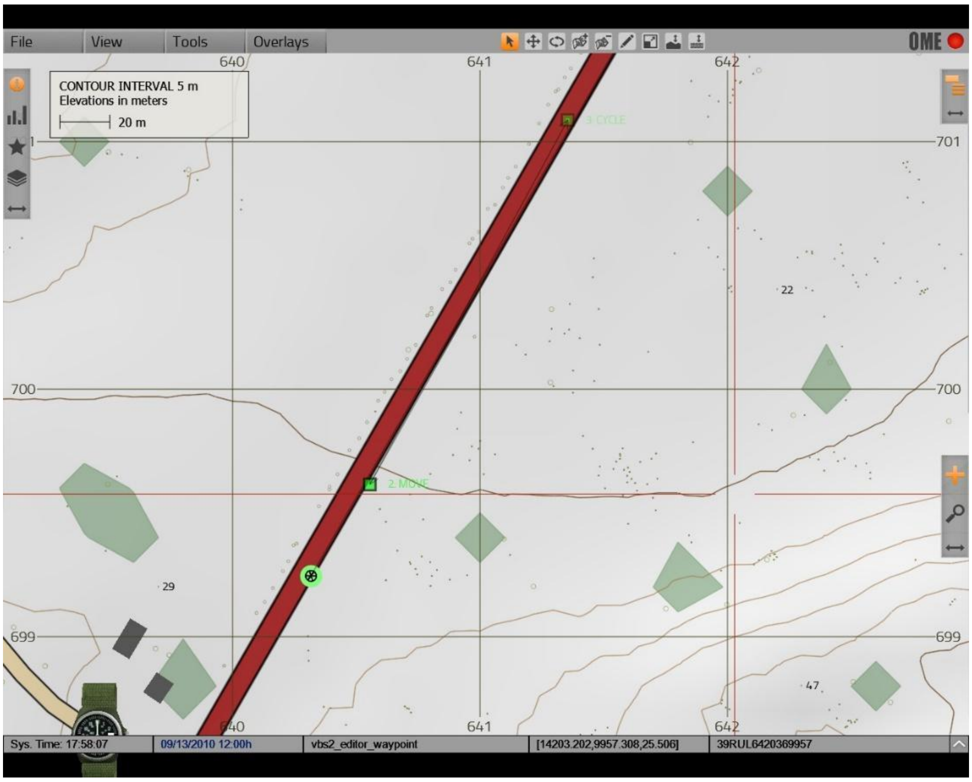


Figure 2.33

Referring to the Fig. 2.33, the truck is represented by a green circle, the Waypoints by a green rectangle.

Looking at the script, the only difference between the script explained before is in the command waitUntil in line 28: **`waitUntil {(damage car_4)>0.1};`.**

In this case, the helicopter will not move until the truck is damaged differently for the cars that had to be destroyed to activate the Trigger and make the helicopter move. The quantity of damage is 0.1. The range of damage goes from 0 to 1 (with 1 the truck is destroyed).

<u>Moving Target: boat, Fig 2.28 quarter (64-65; 67-68), script lines (38-45)</u>

Also in this case the boat moving thanks to the Waypoints with the cycling option.

The commands in the scrip are the same explained before. It is useful only to show some pictures in order to understand how this part looks like.

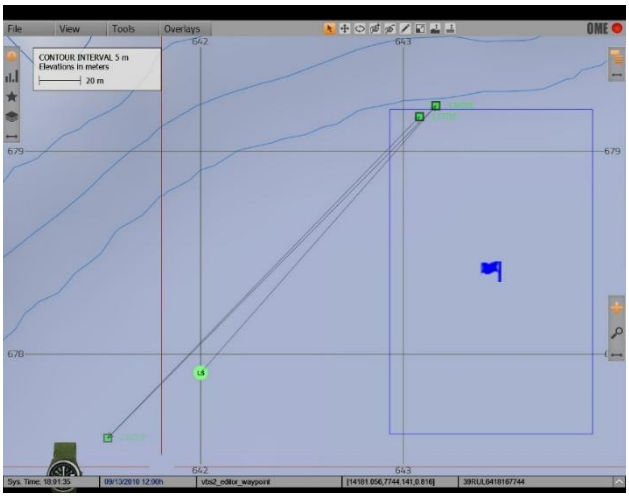


Figure 2.34

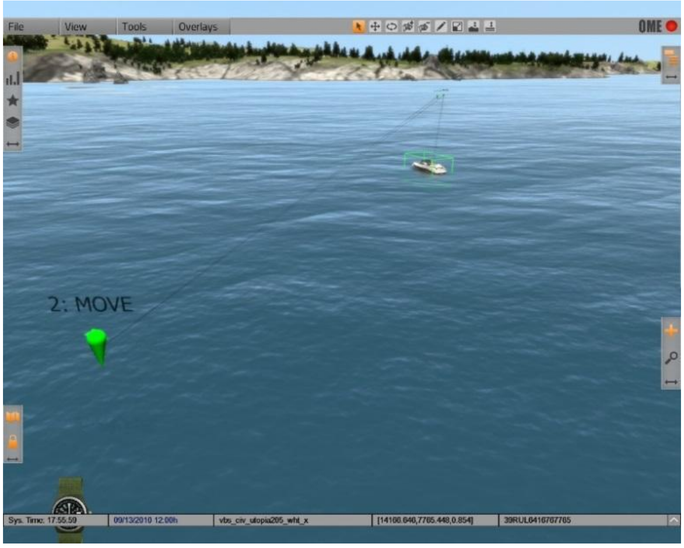


Figure 2.35

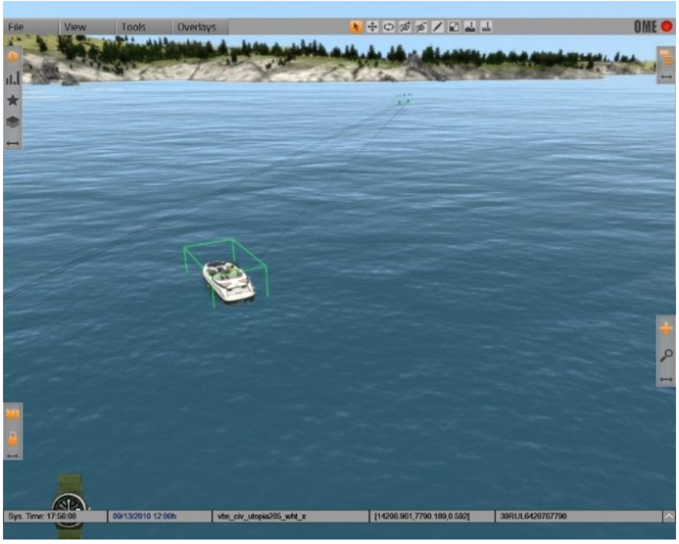


Figure 2.36

In the previous figure is shown the boat with attached the Waypoint. It is shown in the 3D view and maps view.

Also in this case, the boat is represented by the green circle and the Waypoints by the greens rectangles.

<u>Static target: barrel, Fig 2.28 quarter (60-61; 69-70), script lines (47-50)</u>

This is the last target, there is a barrel in the middle of the road and the trainer has to destroy it.

The commands in the script and the implementations in the game have been explained before.

The barrel is little, it is because is the last task and it will be the most difficult. For the trainer is not easy to recognize the barrel.

The barrel is shown in Fig 2.37, from a close view.

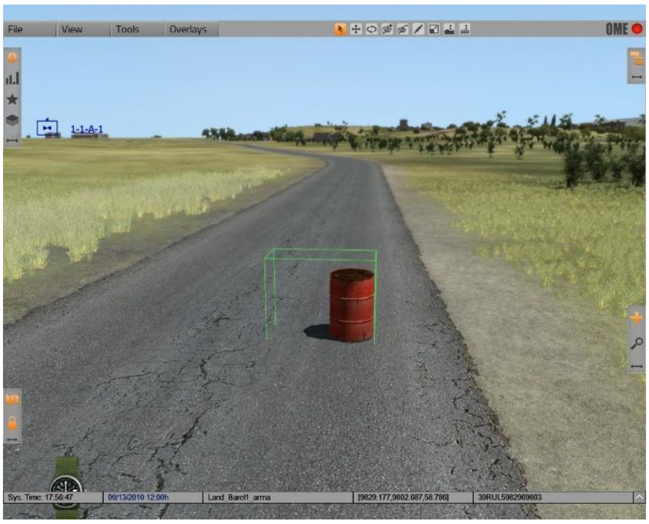


Figure 2.37

<u>End of the simulation: come back to the airport landing, Fig. 2.28 quarter (60-61; 70-71), script lines (51-52)</u>

At the end of the simulation the trainee will be brought back to airport.

In order to make the helicopter land to the last trigger´s position, a Trigger called LAND has been created. The object properties window of this Trigger is shown below:

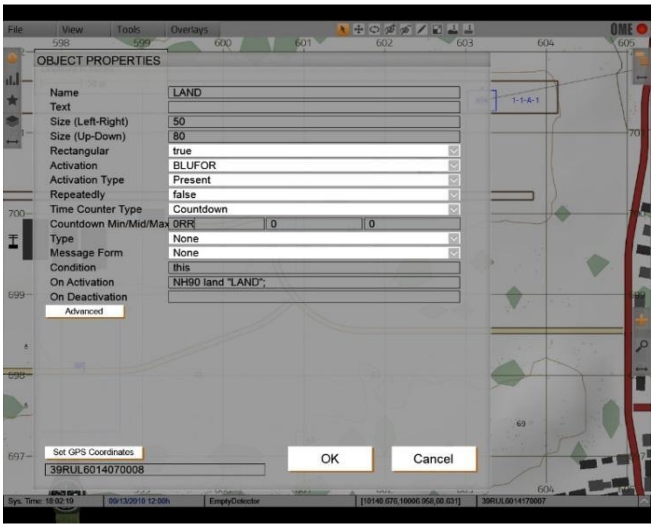


Figure 2.38

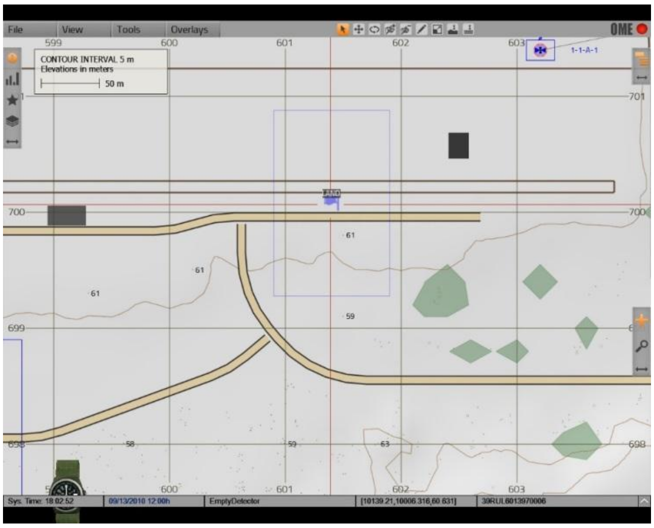


Figure 2.39

The code to make the helicopter land is written in the On Activation field with the following syntax:

NH90 land ‘’LAND’’

- NH90 is the name of the helicopter
- Land is the command that makes the helicopter land
- LAND is the name of the trigger that passes the position of landing to the command land.

2.7 Human Factors – Visual Stimuli Automatic Feedback

Human factors is the study of how people interact with machines and environments. In the past few years, extended reality has emerged as a powerful tool for human factors research. With the ability to immerse users in digital worlds, XR offers a unique opportunity to tailor experiences to suit a variety of human needs and preferences.

Indeed extended reality technology (augmented, virtual or mixed) is a key technology used by researcher to develop methodologies for the study of psychological aspects.

Moreover, the availability of consumer XR devices is expected to raise to 43.5 million by 2025 (Ratcliffe J et al.,2000).

Therefore, it is crucial to study phenomena associated with potential negative effects on users and develop solutions to optimize XR technologies usage on a wide range of consumers.

This chapter focuses on a methodology developed for studying the reaction of trainees to visual stimuli using virtual reality.

In this case a methodology is based on a technical solution whose main parts consists in a virtual scenario created specifically for helicopter crew, the choice of virtual objects designed to create comprehensive visual stimuli and the automatic logging of reliable data.

The scenario has been created using VBS3 3D editor that allows to set object position and the helicopter dynamics, the objects used as visual stimuli are have been designed in a 3D modelling software and imported inside VBS and the logging automation have been implemented coding in the proprietary language of VBS3.

2.7.1 Mission Editor – Virtual Scenario Creation

In figure below is showed how the scenario looks like in the Mission Editor. The helicopter will follow the path clockwise starting from the quarter (60-61;70-71).

The first movement of the helicopter is driven by Waypoint in order to fix its high at 100 ft as explained in the old Documentation in paragraph called Single Player Scenario using triggers.

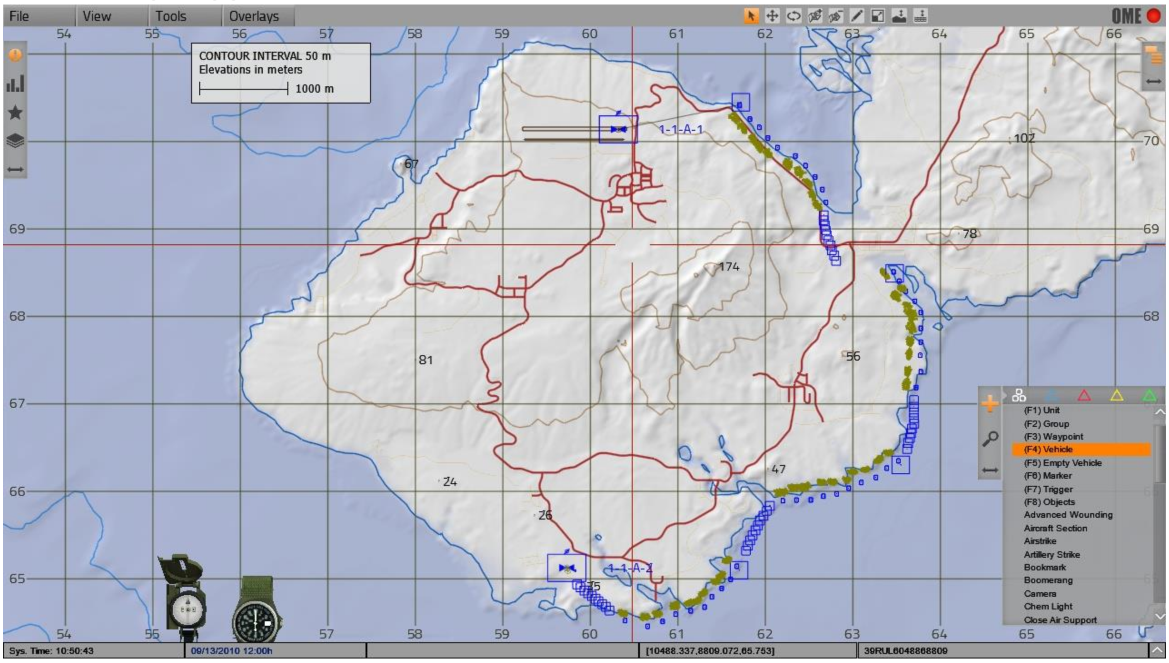


Figure 2.52

<The scenario is designed to be played by the doorgunner side. The candidate will sit during the experiment and will be exposed 80 trials: 40 sets of LR and 40 of sets of spheres.

Figure 2.53

Figure 2.54

LR Sets - Logic:

At the beginning of every set of LR there is a big trigger that prints in the screen these messages:

Do you see a ring that opens up?

OR

Do you see a ring that opens down?

The sintax for printing the message on the screen is:

titleText[“Do you see a ring that opens up?“]; titleFadeOut 8;

titleFadeOut is a command that terminate the title effect and set duration of the fade out phase to the given time.

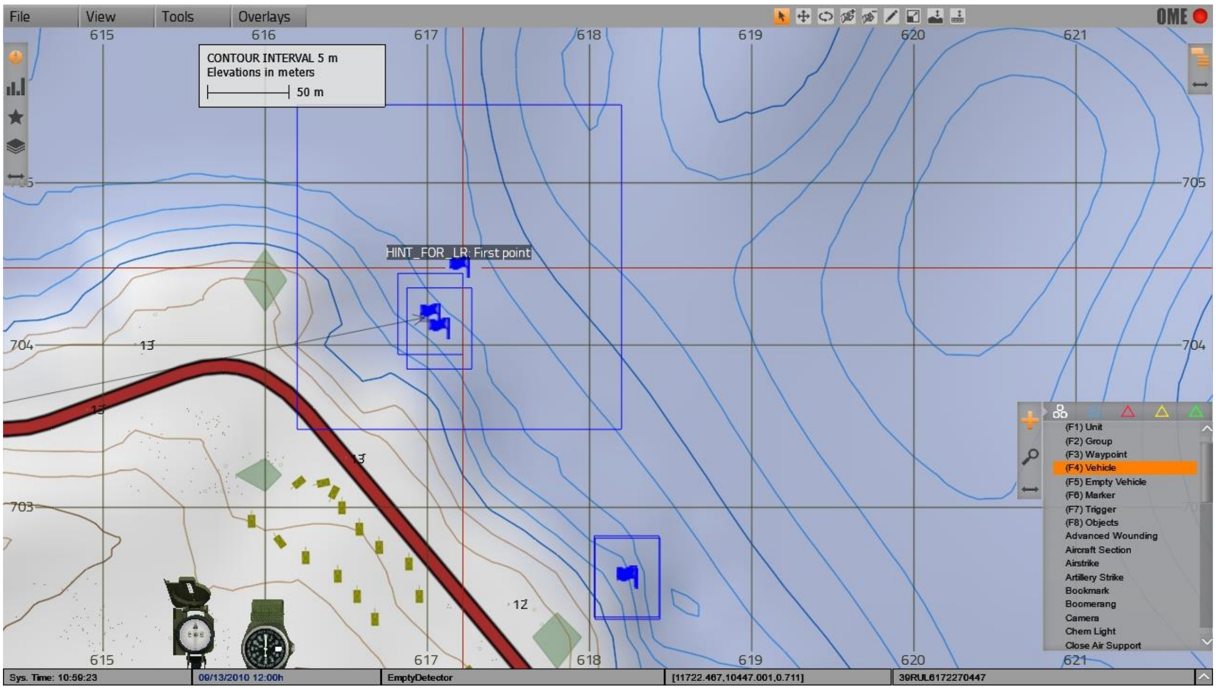


Figure 2.55

Then in front of every set there are two triggers: one for passing the position of the set of LR to the helicopter and one to call the rotation function.

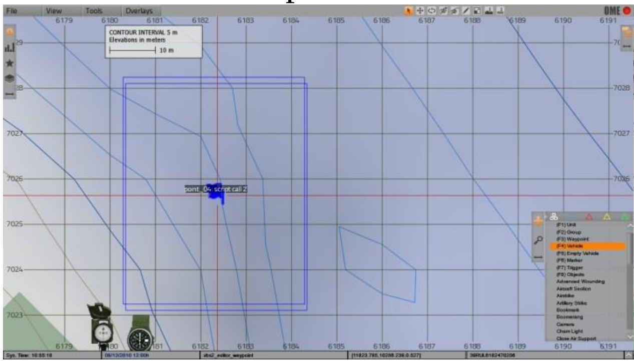


Figure 2.56

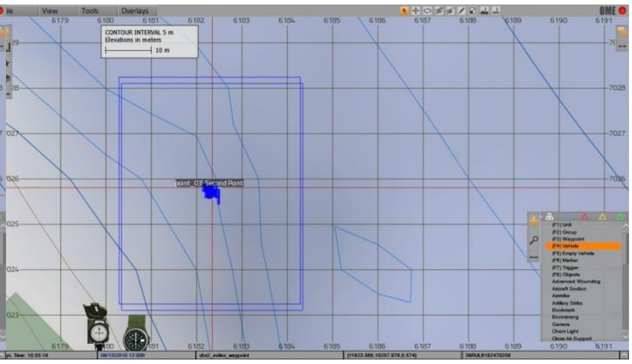


Figure 2.57

<u>Spheres Sets - Logic:</u>

In this case there is only one trigger in front of the set. This is used to print on the screen a question asking how many spheres does the user count. Furthermore, the trigger is used to pass the position and when it is activated it makes the spheres appear. The aim of this task is to have the result whether the candidates can count objects correctly within a defined time frame.

In both cases the generation of this visual stimuli (rings and spheres) has been decided as a requirement by the psychologists collaborating in the project.

2.7.2 Scenario Automation Code

The logical functions that have been developed in this scenario are:

- Objects timing: 3 seconds for LR and 4.5 for Spheres.
- Motion: The helicopter is moved from one trial to the other with a command called doMove that gets the position from the triggers´ position
- LR Rotation: All the LR had to be oriented with an angle that can be seen by the trainee
- Input Mapping: Users wearing a headset might result isolated from the real world. Due to this aspect, it has been necessary to find an easy-to-handle ergonomic device that could be used during the operations. The device chosen is presenter whose buttons are automatically mapped on the corresponding keys on the key board. The two buttons present on the keyboard map respectively a positive and negative answer.
- User´s response time: The time from the moment in which the obejcts appear to the moment in which the trainee presses a key in the input device. This has been done by printing a starting time and a finishing time for every trial in the Log file of VBS3.

The scripts that has been written in order to implement the described functions are:

- Init.sqf: this script is the Initialzation Script. It runs every time that the simulation starts.
- Motion_Script.sqf: This script has different functionalities:
    - It moves the helicopter from one trigger to the other
    - It enables the input from the Presenter
    - It makes appear and disappear the sets of Spheres (for the LR it is done in the Rotation function)

    - It starts and print the time in the Log File for the set of Speres (for the LR it is done in the Rotation function)
- Rotation1.sqf to Rotation40.sqf: This 40 scripts are called once per every set of LR.
  These scripts orientate the sets of LR with an angle that allows the trainee that is sitting at the crew position to see all the LR.
  They start and print in the Log file the at the time when the sets appear.

- YES_STOP_BUTTON.sqf & NO_STOP_BUTTON.sqf: This two scripts are called when the trainee presses respectively the right or the left button on the presenter.
  They print the at the time when the trainee presses the button in the Log file and they will indicate if the aswer if positive or negative.
  Subracting the Starting time (that comes from the Motion_Script.sqf for the sets of spheres, and from the Rotation scripts for the sets of LR) with the Stop time that comes from these two scripts the User´s response time is obtained.

Below there will be reported in blue the code implemented in order to perform such operations:

2.7.3 Init.sqf

This paragraph describes the implementation of the init.sqf function.

```
execVM "Motion_Script.sqf";
```

The command is used to run a script, since this command is inside the Initialization script, it will be run at the start of the simulation.

```
Array =[LR_11, LR_12, LR_13, LR_14, LR_15, LR_16, LR_17, LR_18, LR_19, LR_110,
LR_111, LR_112, LR_113, LR_114];
```

This is the first array of LR: it is important to call the LRs with the same name used in the Mission Editor (the same for the sets of Spheres).

```
{_x hideObject true;} forEach Array;
```

The command forEach cycle an instruction on an Array, in this case it is used to hide all the LR present in the map at the start of the simulation.

```
ArraySp_1 = [Sphere1_1, Sphere1_2, Sphere1_3, Sphere1_4, Sphere1_5, Sphere1_6, Sphere1_7,
Sphere1_8, Sphere1_9, Sphere1_10, Sphere1_11, Sphere1_12];
{_x hideObject true;} forEach ArraySp_1;
```

2.7.3 Motion Script

As mentioned, this function implements several logics (helicopter dynamics, users logging and object timing for positioning). There are reported below the code and comments necessary for the implements:

-Logic for sets of LRs:

```
////////////////////////////// Timing Lines
waitUntil {(triggerActivated point_01)};
```

This command blocks the procedure until the trigger called point_01 is activated.

```
_keyDown = (findDisplay 46) displayAddEventHandler ["KeyDown", "if (_this select 1 == 209) then
{execVM 'YES_STOP_BUTTON.sqf'}"];
```

```
_keyDown = (findDisplay 46) displayAddEventHandler ["KeyDown", "if (_this select 1 == 201) then {execVM 'NO_STOP_BUTTON.sqf'}"];
```

These two lines contain the syntax that is necessary to get an input from the keyboard in VBS3. In this case 201 is the codification for Bild up and 209 for Bild down. Once they are pressed in the keyboard or in the Presenter, the scripts 'YES_STOP_BUTTON.sqf' or 'NO_STOP_BUTTON.sqf' are called.

```
waitUntil {(triggerActivated point_00)};
sleep2;
NH90 setVelocity [(vectorDir NH90 select 0)*35,(vectorDir NH90 select 1)*35,(vectorDir NH90 select 2)*35];
```

This command sets the Velocity of the Helicopter along the three components.

```
NH90 doMove getPos point_03;
```

This command is used to move the Helicopter from the actual position to the position of the trigger called point_03.

-Logic for sets of spheres:

```
ArraySp_1 = [Sphere1_1, Sphere1_2, Sphere1_3, Sphere1_4, Sphere1_5, Sphere1_6, Sphere1_7, Sphere1_8, Sphere1_9, Sphere1_10, Sphere1_11, Sphere1_12];
waitUntil {(triggerActivated point_21)};
{_x setScale [5,5,5]} forEach ArraySp_1;
```

This command will scale the Spheres 5 times in each direction, making them appear 5 times bigger.

```
{_x hideObject false;} forEach ArraySp_1;
```

This command will make the spheres appear.
The following block of lines is the one that will print the at the time when the spheres appeared in the scenario.

```
_startSphere = diag_tickTime;
```

diag_tickTime is the command that prints the time in the Log File.

```
_beginSphere = ";SPHERE;START;";
_semicolon = ";";
_motionScript = "Motion_Script";
```

These three string variables are used to have a string separated by semicolon in the Log file. It is necessary to have semicolons that separe every word/number in order to import the datas in Excel.

```
diag_log format ["%1 %2 %3 %4 %5", _beginSphere, _startSphere, _semicolon, _motionScript, _semicolon];
```

This command prints the string in the log file.

```
uiSleep 4.5;
With this command the spheres will appear for 4.5 seconds.
{_x hideObject true;} forEach ArraySp_1;
```

This command makes the spheres disappear again.

```
_keyDown = (findDisplay 46) displayAddEventHandler ["KeyDown", "if (_this select 1 == 209) then {execVM 'YES_STOP_BUTTON.sqf'}"];
_keyDown = (findDisplay 46) displayAddEventHandler ["KeyDown", "if (_this select 1 == 201) then {execVM 'NO_STOP_BUTTON.sqf'}"];
```

2.7.4 Rotation Function

The rotation scripts are called by triggers located in front of the LRs sets. This function is fundamental for the system to automatically orientate objects toward the trainee. The requirement for the algorithm to work are:

1. Identify the direction of the users' gaze in the virtual world: obviously the users are free to move their heads freely during the simulation. Therefore, the system has to automatically recognize the orientation of the user head when the script is triggered
2. Assign the correct orientation to the rings: the position retrieved from the user has to be on the same direction of the user gaze but on the opposite verse
3. Rotate the rings toward the trainee: in order to do so the ring has to rotate on his vertical axis and maintain the position assigned during the scenario modeling.

Here below it is reported the technical implementation of the function in term of code and comments:

```
_PlayerDir = vectorDir player;
```

vectorDir is a command that return the vector [x,y,z] representative of the players direction at the time when the script is called.

```
_xpl = _PlayerDir select 0;
```

```
_xplneg = -_xpl;
_ypl = _PlayerDir select 1;
_yplneg = -_ypl;
_zpl = _PlayerDir select 2;
_zplneg = -_zpl;
```

These three lines are used to invert the vector of the player in order to have the opposite vector and then assign it to every LR of this set.

```
_PlayerDir1 = [_xplneg,_yplneg,_zplneg];
```

This is the array that contains the player´s opposite vector.

```
Array =[LR_11, LR_12, LR_13, LR_14, LR_15, LR_16, LR_17, LR_18, LR_19, LR_110, LR_111,
LR_112, LR_113, LR_114];
{_x setVectorDirAndUp [_PlayerDir1,[0,0,1]];} forEach Array;
```

The command setVectorDirAndUp assign the vector _PlayerDir1 to every element of the array called Array. This array is simply a list of all the LRs present in this trial.

```
{_x setScale [5,5,5]} forEach Array;
{_x hideObject false;} forEach Array;
_start = diag_tickTime;
_begin = ";;START;";
_semicolon = ";";
_rotation = "Rotation1";
diag_log format ["%1 %2 %3 %4 %5", _begin, _start, _semicolon, _rotation, _semicolon];
sleep 3;
{_x hideObject true;} forEach Array;
```

These last lines have exactly the same function of the lines described for the Spheres in the Motion_Script.sqf. They scale the LRs, make them appear, start the time and print it in the Log_file and the make them disappear again.

2.7.5 Positive Answer Log

This script is called when the right button of the presenter is called.

```
_stop = diag_tickTime;
_finish = ";;NO_STOP;";
_semicolon = ";";
_yesScript = "YES_STOP_BUTTON";
diag_log format ["%1 %2 %3 %4 %5", _finish, _stop, _semicolon, _yesScript, _semicolon];
```

This script uses the same explained in Rotation1.sqf and in the Motion_Script.sqf for printing the starting time in the Log File.

They are different because they print a Stop time and in this case a negative answer from the trainee.

2.7.6 Negative Answer Log

This script works with the same logic of YES_STOP_BUTTON.sqf but it return a positive answer.

This line of code is useful to map the codification of keyboard in VBS3 :

```
waituntil {!(IsNull (findDisplay 46))};
_keyDown = (findDisplay 46) displayAddEventHandler ["KeyDown", "hint str _this"];
```

It print on the screen the number of the key that is being pressed.

There is a demo mission running this command called RINGSORIENTATION_KEYPRESSED.intro. The mission can be opened and run in the Mission Editor.

2.7.7 Data Generation

The log file of VBS3 contain heterogeneous data coming from the system operations. In order to analyze and extract the valuable data the log file analyzes it through a macro containing a algorithm that cancel the undesired data and extract the inputs from the trainees:

The procedure used in order to get VBS3 log file is the following:

- Open the Log file in Notpad++, Log File name: VBS3_64.RPT
- Connect the presenter to the computer and turn it on
- Launch VBS3 leaving NotePad++ in background
- Open the desired Mission
- Run the Mission and write down the clock time
- Once the mission is finished Abort the Mission
- Go to the Notepad++ Window
- Search the following line:";;START; 1234.56789 ; Rotation1 ;": this line identify the start of the logging
- Select everything that is under this line, copy it and paste it to a new file.
- Save the file as a .txt.

<u>Log File Results:</u>

This section presents the log file results extracted with the methodology presented in the previous chapters.

```
11:53:57 "START:, 3940.6591796875"
11:54:07 Trying to move object  VBS3\people\nz\nzdf_pilot\nzdf_pilot_heli.p3d which is moved out. setSimulationMode 2 was probably called on attached object from script which is not correct.
 -- <Last message repeated 8 times>
```

11:54:09 Warning Message: No entry 'bin\config.cpp\CfgVehicles\vbs_us_af_ac130h_gry_l60_gau12_m102_x\scope'.
11:54:09 Warning Message: '/' is not a value
11:54:09 Warning Message: No entry 'bin\config.cpp\CfgVehicles\VBS3StrikeAC130_IR_VEHICLE\scope'.
11:54:09 Warning Message: '/' is not a value
11:54:09 Warning Message: No entry 'bin\config.cpp\CfgVehicles\vbs_us_af_ac130h_gry_l60_gau12_m102_x\scope'.
11:54:09 Warning Message: '/' is not a value
11:54:09 Warning Message: No entry 'bin\config.cpp\CfgVehicles\VBS3StrikeAC130_IR_VEHICLE\scope'.
11:54:09 Warning Message: '/' is not a value
11:54:35 Trying to move object VBS3\people\nz\nzdf_pilot\nzdf_pilot_heli.p3d which is moved out. setSimulationMode 2 was probably called on attached object from script which is not correct.
-- <Last message repeated 8 times>
11:55:11 "START:, 4014.31127929688"
11:55:12 " _YES_STOP:, 4015.453125"
11:55:15 " _NO_STOP:, 4018.4052734375"
11:55:20 "START:, 4023.99609375"
11:55:22 " _YES_STOP:, 4025.9482421875"
11:55:22 " _YES_STOP:, 4025.9482421875"
11:55:30 "START:, 4033.30810546875"
11:55:32 " _NO_STOP:, 4035.31420898438"
11:55:32 " _NO_STOP:, 4035.31420898438"
11:55:32 " _NO_STOP:, 4035.31420898438"
11:55:35 " _NO_STOP:, 4038.55908203125"
11:55:35 " _NO_STOP:, 4038.55908203125"
11:55:35 " _NO_STOP:, 4038.55908203125"
11:55:37 " _YES_STOP:, 4040.869140625"
11:55:37 " _YES_STOP:, 4040.869140625"
11:55:37 " _YES_STOP:, 4040.869140625"
11:55:45 "START:, 4048.05224609375"
11:55:46 " _YES_STOP:, 4049.16625976563"
-- <Last message repeated 4 times>
11:55:56 "START:, 4059.0380859375"

```
11:55:58 " _NO_STOP:, 4061.33813476563"
11:55:58 " _NO_STOP:, 4061.33911132813"
 -- <Last message repeated 4 times>
11:56:09 "START:, 4072.22119140625"
11:56:10 " _NO_STOP:, 4073.8623046875"
 -- <Last message repeated 6 times>
11:56:12 " _YES_STOP:, 4075.46118164063"
11:56:12 " _YES_STOP:, 4075.46118164063"
11:56:12 " _YES_STOP:, 4075.46118164063"
11:56:12 " _YES_STOP:, 4075.46215820313"
11:56:12 " _YES_STOP:, 4075.4931640625"
11:56:12 " _YES_STOP:, 4075.4931640625"
11:56:18 "START:, 4081.17529296875"
11:56:20 " _NO_STOP:
```

The data prepared from this procedure are ready to be analyzed by the subject matter expert (i.e. human factors scientists, psychologists, engineers).

Indeed, the result of this methodology will provide data related the user attention. Analysts will have the result of the user capabilities of recognizing object orientation and number during the simulation and the user response.

The user response is obtained subtracting the start and finish time in the log file.

CHAPTER 3

Internet of Things and Cloud Technologies

3.0 Overview

Nowadays computing power, software products, bandwidth and storage are rapidly enhancing in term of capabilities and cost. These factors are enablers to the development of innovative technologies that require connectivity and communication between heterogeneous sources. Internet of Things is a set of technologies that takes advantage of this context. Indeed, the purpose of IoT is to connect the physical and the virtual world in order to obtain insights and improve complex processes. IoT can be applied to a wide set of sectors. Several projects have been developed in order to connect machine at the shopfloor, vehicles, traffic control, smart cities, energy networks, people (smart devices) etc.

Indeed, the adoption of IoT based on a cloud infrastructure introduces and end-to-end transformation to the ecosystem they are applied. From a technical point of view, the first step is often represented by the creation of reliable data sources, sensors connected to the field feed databases that support analytics systems enabling companies, agencies and institutions to understand deeper the processes shortfalls and creating historical databases to be used to design improvement strategies.

Furthermore, IoT paved the way to new strategies for the development of new business models (i.e. pay per use). It is interesting to investigate the aspects of this technology that relate to the system architecture and the main use cases, identifying the whole stack of technologies deployed - from data collection to stream analytics and optimization strategies.

This chapter will focus on the technological aspect of this thread of digital innovation identifying the characteristics and challenges. The use cases that will be presented in Chapter 4 focus on the application of IoT to the energy optimization. In particular it will be presented how this technology enables the optimization of all energy carriers to a high granularity enabling complex improvement strategies. Furthermore it will be presented a work on cloud manufacturing presenting a commercial architecture to achieve automatic recognition of digital documents. The figure below gives a glance at the IoT adoption by sector.

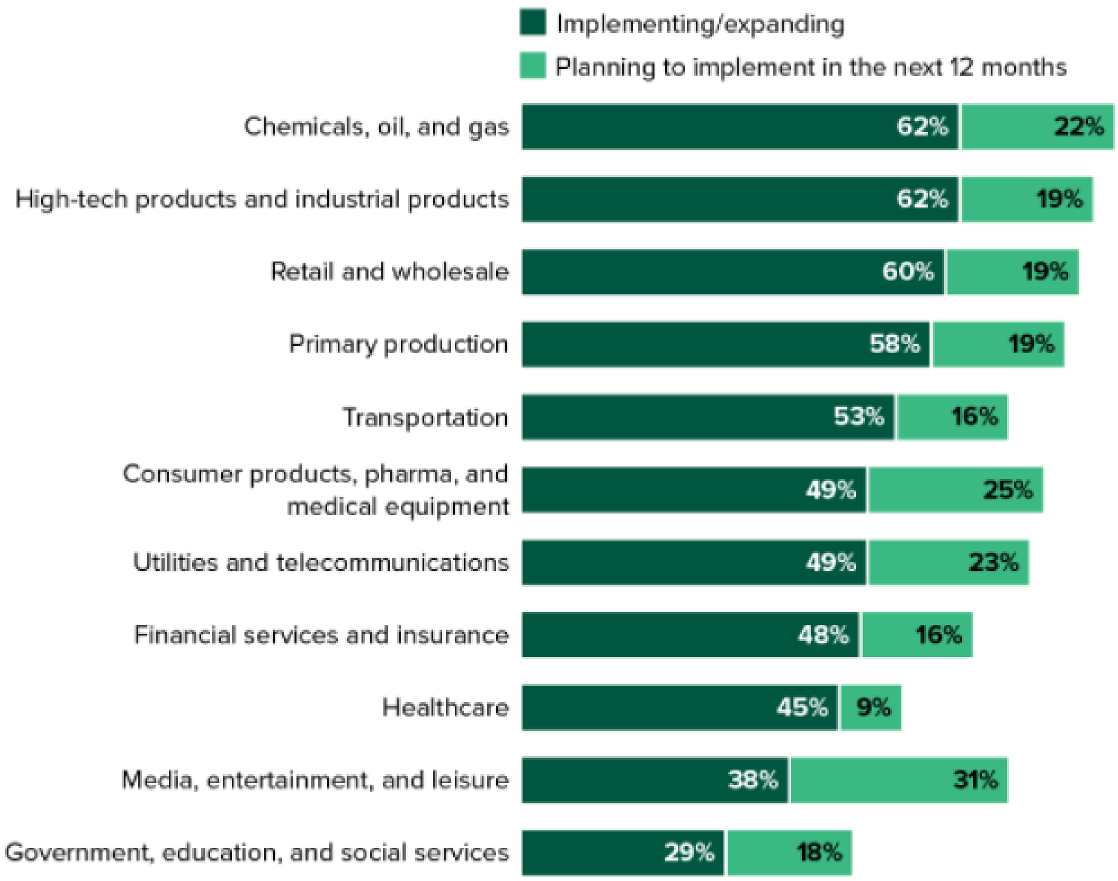


Figure 3.1 – IoT adoption by sector – source: "Internet-Of-Things Heat Maps For Operational Excellence In 2021" - Forrester

The figure above is interesting to observe that this technology has had a quick adoption in sectors related to complex processes, but this trend is quickly spreading in all sectors where the users have different characteristics.

3.1 IoT Technology Introduction

IoT can be defined as a network of sensors, devices and systems that establish an interaction in order to retrieve data and events.

Traditional internet connection enables interaction between a limited number of devices and people, Internet of Things instead, aims to connect different entities and create a reliable network to establish relationships between heterogeneous devices without the direct intervention of humans.

IoT infrastructures allow the realization of wireless communication that enable physical systems to stream in real time valuable data. Furthermore, wireless devices for different purposes (energy metering, biometric sensors, etc.) are getting more and more common and cheap. Indeed, sensors are getting more and more embedded in systems and frictionless in terms of information collection. For example, data related to energy consumption, can be retrieved with non-intrusive technologies (i.e. current clamp), smart cameras equipped with neural network accelerators analyze video in real-time and biometric data can be collected just tapping a sensor.

IoT is a versatile technology, and the same concepts can be applied to different application.

Figure 3.2 – IoT Application Domain - Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications (Ala Al-Fuqaha et al).

In order to realize such systems, it is necessary that the development of technologies and applications grow at the same pace of the market and users' requests.

Protocols and devices are required to evolve respectively in term of general communication between different entities and usability. Furthermore, architecture must converge into these new standards to create an environment where customers can reap the best value to develop solutions.

Moreover, architecture design is a fundamental part to consider. Indeed, it is often necessary to find trade-off between defining a standard architecture for the system to be scalable and open to integrate with different systems and the constant revision to match the IoT limitations.

In fact, as the number of connected devices continuously increase it is not trivial to consider all the underlying protocols. For this reason, it is a good practice to use a large addressing space (i.e. IPv6). Furthermore, it is important to consider the aspects related to privacy and security, especially when the IoT network is capable of monitoring and controlling physical objects. Finally, It has to be considered that in an industrial context IoT systems must ensure a real impact on client at an efficient cost.

3.2 IoT Market Characteriscs

The number of connected devices is expected to overtake 17 billion in 2030 and generate a revenue associated with electronics vendor of $762 billion (Peter Middleton et al.,2022).

Since the implementation of IoT systems involves the hardware, software products and IT services, it is changing the IT/OT market it started in. Indeed, in many cases IoT platform are nowadays starting to enhance or also replace legacy systems and as as an example of IT/OT convergence, IT companies are enhancing their products with IoT functionalities (i.e. data contextualization, etc.) and Operation technology vendors are adding digital application to IoT systems.

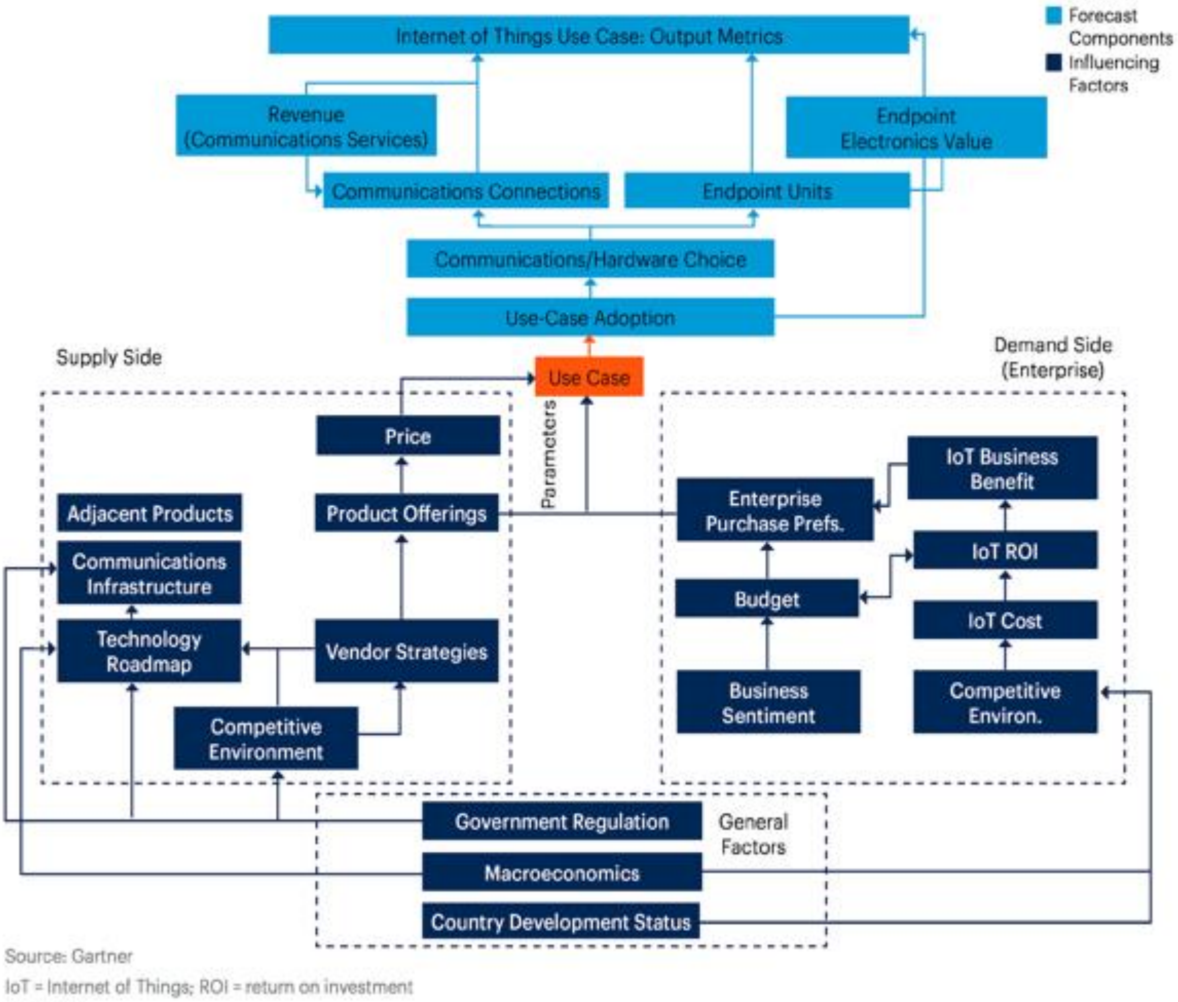


Figure 3.3 – IoT Market Characteristics

Healthcare and manufacturing applications are expected to impact the most the economics of the IoT ecosystem. Application related to healthcare like mobile health and remote care in order to achieve prevention, treatment and monitoring system are projected to create about $1.1-$2.5 trillion in growth annually by 2025 and the whole annual impact is estimated to reach $6 trillion (J. Manyika et al, 2020).

These projections give an opportunity for manufacturers to transform their products ensuring them to be smart and connected with new technologies and able to communicate in a complex environment to extract meaningful insights to enhance control and design improvement strategies.

3.3 State of Art Architecture

In order to take advantage of IoT it is crucial to define an architecture that enables companies to get high-quality data and standardize data coming from heterogeneous sources. IoT systems must be capable of connecting several systems and so it is clear the need of having a flexible architecture.

Depending on the use case architecture can change. Below it is reported a reference architecture representing the fundamental elements that constitutes an IoT systems.

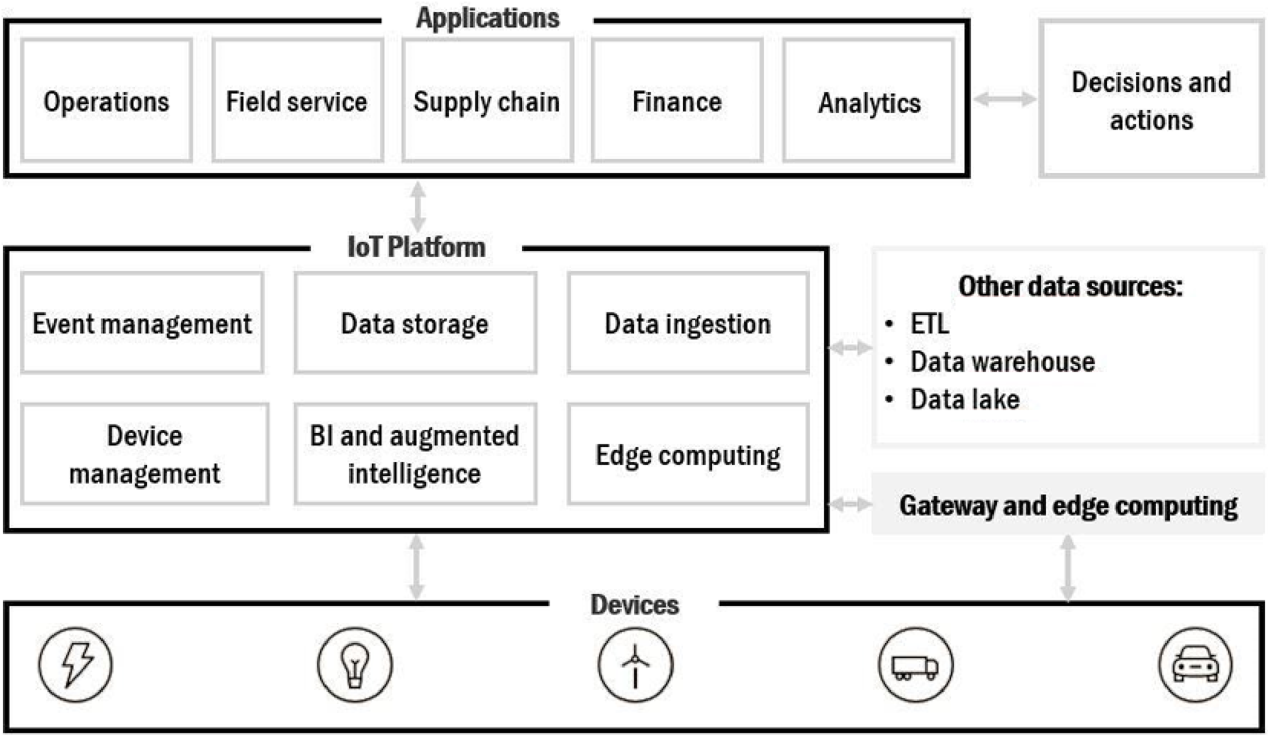


Figure 3.4 – Reference Architecture

Even if the requirements of each system have to be analyzed based on the goal and characteristics, recent literature identifies main components that build up typical architecture of IoT systems.

The approach used to describe the components logically follows the figure above reading it from the bottom. Before describing each component in more details it is reported a quick list of the five main components (or sections) that have been chosen in order to describe a high-level conceptualization of IoT systems:

- Devices and data sources
- Data Transfer
- IoT Platform
- Application Layer
- Decision Layer

3.3.1 Devices and data sources

This part of the architecture is the one that refers to the devices or third-party systems with the purpose of generating a distributed and reliable data source.

This section contains sensors and systems with different scopes, they can be applied for retrieving several types of data like physical data (temperature, pressure, vibration, etc), environmental data (weather conditions, humidity, etc.), context information (building occupancy, production order, production item, shift, etc.), biometric data (heart rate, blood pressure), etc.

In order to standardize the data flow of heterogenous entities, the market offer some plug-and-play solution that provide a good solution especially when operating in a green field. Sometimes when the requirement needs a special solution, scientist and developers build custom application to standardize the data. Indeed, it is possible to implement plug-ins that translate diffent protocols and provide standardized data.

In the last few years machine learning has entered also this stack of the architecture. For example a component that evolved the analysis of video analysis is represented by the neural network accelerators that enables on-edge to extract automatically useful

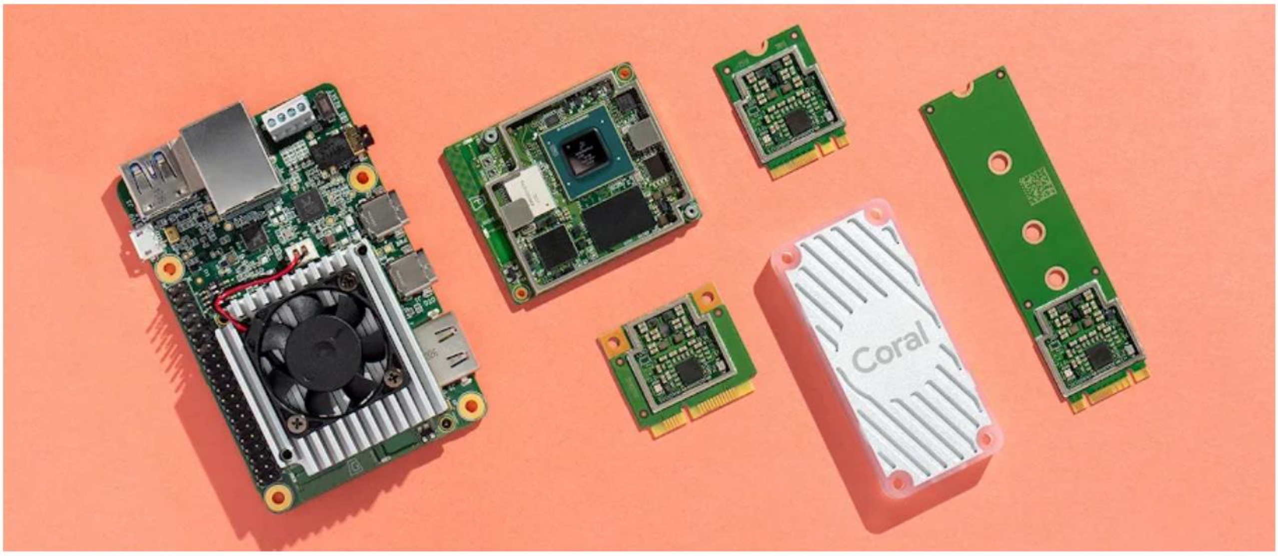


Figure 3.5 – Google Choral

information without compromising users' privacy. A commercial example is the neural network accelerator developed by Google called the Google Choral.

Indeed, for example this toolkit provides on-edge inferencing capabilities and allows engineers and developers to design products that are off-line. Furthermore, it respect users privacy performing inferencing locally. Currently, the range of commercial and experimental sensors is very wide, this work will present in Chapter 4 more sensors dedicated to the monitoring of energy carriers.

3.3.2 Data Transfer

This stack of the architecture is dedicated to transfer the object to the upper stack securely. The main technologies used to transfer data are RFID, GSM, UMTS, Wifi, etc.

Also cloud technologies and data management functions can be used to handle this layer.

3.3.3 IoT Platforms

These tools are used to enable engineers to work with several objects without worrying about a specific hardware platform. Moreover, IoT platforms can be cloud based or on-premises and their main scope is to make operation on retrieved data (processing, etc.), delivers service over the network and manage, control various types of endpoints. They normally provide infrastructures to support the implementation of logics and applications for enabling operations.

3.3.4 Application Layer

This section is designed in order to deploy specific functionalities to the system users. Application layer can be made up of different components designed for different purposes.

Indeed, IoT data can integrate enterprise system with context information or feed custom application to satisfy specific use case requirements.

The use cases can be developed for different purposes, depending on the goal of the application, the application layer can deploy real-time analytics and/or historical analysis. Furthermore, it is a key decision to find the correct solution related to a cloud or en-edge application.

A use case focused on the conceptualization of an energy management tool that mostly rely on this layer will be presented on the last chapter of this thesis.

3.3.5 Decision Layer

The decision layer is the most high-level section of the reference architecture. This layer is designed in order to provide to users clean and comprehensible information to humans.

Depending on the user characteristics (production, operations, top management, , etc.) this layer build different functionalities, generally its main goal is to give at a glance relevant information that enable user to make fast and smart decision or conduct more in-depth analysis.

This layer can be used to deploy dashboard, flowcharts, graphs and more visual information elaborated from the application layer.

From a developer point of view, it can also be a tool that enables new IoT components.

In the last few years big data and machine learning/artificial intelligence solutions are being applied to this layer. On the other hand, this layer can be used to balance the signals received by the systems and the one expected to maintain and increase the system performance.

3.4 IoT Building Blocks

It is interesting to have an overview of the components that build and IoT system in order to grasp better the real meaning and functionality of IoT.

In literature six main components are generally identified:

- Identification
- Sensing
- Communication
- Data Processisng
- Services
- Semantics

3.4.1 Identification

It is important that in an IoT infrastructure to match the services with their request. The aim of this techniques is to give to IoT object a unique identification.

There are several ways of matching that can be deployed, it reported below a list with some examples:

- IPv4, IPv6: these are both IP addresses characterized by a binary numbers. These addresses can be used to identify machines connected to a network. The main differences are: IPv4 is a 32 bit binary number while Ipv6 is 128 bit and IPv4 is separated by periods while IPv6 are separated by colons.

- 6LoWPAN: this is an acronym that stands for Low-Power Wireless Personal Area Network. It is firstly deployed to enable devices with limited computing capabilities to be part of the Internet of Things. This devices are usually deployed in order to perform monitoring tasks both in industrial production, smart grid and smart homes.

- RFID: Radio frequency Identification, its main application in IoT cover a crucial area of applications related to track and trace. Indeed, with RFID technologies it is possible to automatically track objects within a defined space, it is common to see these application in warehouses and shopfloors

- Near Field Communication: it enables the communication of close objects (0 to 2 cm), it became very popular being embedded in smartphones and used for contactless payments. It is commonly used in industrial environments

- Electronic product code (EPC): the EPC structure can be found in the EPC global Tag Data Standard. They are mostly encoded using RFID.

Moreover, when designing an IoT system it is important to differentiate between entities address and names. Indeed, addresses are referred to the communication network while the name is referred to a particular sensor.

3.4.2 Sensing

In this context, sensing refers to collecting data that are connected in the IoT system and storing them in a database, both on a datawarehouse or on a cloud infrastructure. The data are the basic component to provide users valuable functionalities defined on specific use cases.

As mentioned before, IoT can relate to different entities and the sensors choice is broad. In an industrial context the data can be collected from smart meters, sensors etc. but also directly from the PLCs that control the process automation.

For other use cases also sensor board computers (SBCs) like Arduino, Raspberry PI, etc. are commonly used to build IoT products. Indeed, they easily integrate several interfaces that are useful in context like home automation.

3.4.3 Communication

The communication in IoT systems accomplish the connection of heterogeneous sources to different layers of the system architecture. The communication in such systems is usually carried out using low power, especially when the links are in a disturbed environment. To realize IoT systems there are several protocols whose detail will be further discussed in the next chapters. Communication technologies (some presented in the previous chapter) are based on different operating principles. For example, the RFID that is build up of a tag that is made of a chip attached to an object and a reader. The reader sends a signal to the tag that sends it back and the data is then stored into a database.

The tags can be active when they have a battery, or passive when they just include a chip and do present a battery.

Another common communication technology is represented by the ultra-wide band communication.

This technology is used when the use cases require to connect entities that are growing in real time. Also, WiFi is commonly used, especially for applications where sensors are connected directly to power and have to exchange many data, for example video analysis.

3.4.4 Data Processing

The data processing in an IoT system is carried out by several components, both hardware like microcrontrollers, microprocessors etc. and software products. Single board computers are very common to develop IoT solutions both for prototyping and operating solutions, some examples are:

- Raspberry PI: this is a single board computer; due to its reduced cost it is commonly used also in education but there are some use cases also in manufacturing. These chips are very popular to develop IoT solutions and they are very versatile to realize different use cases.

Figure 3.6 – Raspberry PI

- Arduino: This is one of the famous open-source prototyping platform. It is able to read heterogeneous inputs (sensors, actuators, social network message, etc.) and convert it into an output (motor, publish online data, etc.)

Figure 3.7 - Arduino

- ESP8266: This integrates a 160 MHz microcontroller with a WiFi front-end (client and access point) and TCO/IP with DNS. This chip is popular because of its low price and the integration with the ESP-01 module made by AI-thinker. This provides easly capabilities to embed WiFi and to integrate with other systems or as stand-alone with minimal space requirements.

Figure 3.8 – ESP 8266

- Intel Edison: This a small development platform instegrating a Intel Atom CPU and 32 bit Intel Quark microcontroller. It provides WiFi and mluetooth modules embedded in the chip, together with other feature like UART, SPI, I2C, I2S, GPIO and SD card.

Figure 3.9 – Intel Edison

- Intel Galileo: this is a microcontroller on the Intel Quark SoC X1000 processor and 32-bit system on the chip. It is designed in order to be compatible with Arduino shields.

Figure 3.10 – Intel Galileo

- BeagleBone: This are open-hardware and open-software computers made in order to be as flexible as possible to be plugged in several systems. This SBC is not as versatile as the Raspberry PI but it is really useful for IoT projects with 46 pn header and PRU (programmable real-time unit).

Figure 3.11 - BeagleBone

- Banana Pi BPI-R2 Pro: this represents a good alternative to the Raspberry Pi Compute Module. It is based on SoC Rockchip PK3588, with four core configurated on Dynamic IQ. The form factor is similar to the raspberry PI but this chip has a faster processor and RAM power.

Figure 3.12 Banana Pi

- NodeMCU Dev Kit: This is a development kit based on ESP8266 WiFi chip. It integrates GPIO, PWM, 1-Wire and ADC. It can be easly integrated with NodeMCU that is an open-sorce firmware to enhance the development process.

Figure 3.13 - NodeMCU

- Flutter: This chip is characterized by a ARM processor and provides long-range wireless communication, making it a good choice to develop wireless sensor networks and consumer electronics.

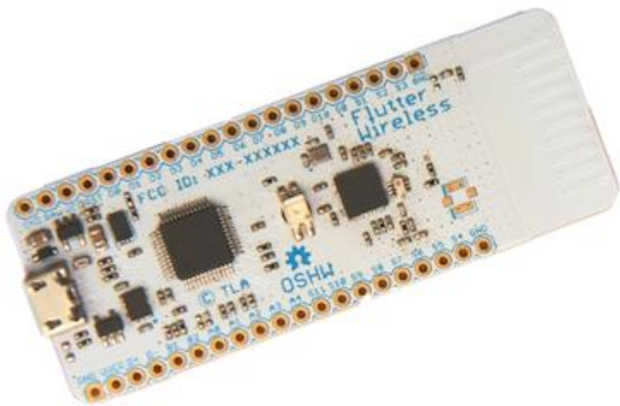


Figure 3.13 - Flutter

As showed before the chip and single computer boards are constantly evolving and private companies and researchers are continuously developing new concepts. The trade-off to choose the correct hardware is often regulated by several factors (i.e. cost, scalability, size, environmental conditions, etc.).

Moreover, many software solutions are developed to provide IoT functionalities. Amid platforms play an important role in the development, operating systems (OS) are a crucial element.

Indeed, OS are the component that regulates the activation time of devices. It is common that IoT devices do not provide sufficient RAM to support the OS. For this reason real time operating system (RTOS) are considered for IoT devices. Indeed, RTOS software can implement more responsive processing for a pre determinate tasks than OS. Also in this field, there are many real-time operating system that can be utilized in order to develop IoT solutions.

Some RTOS also include simulators that allow developers to study solutions emulating IoT systems and wireless sensors network applications.

Some examples are:

- TI RTOS
- Microsoft Azure – Thread X
- SCIOPTA
- Lite OS
- Nano-RK
- Raspbian
- Lepton
- FreeRTOS – now owned by Amazon
- Etc.

Also Cloud Platforms are a fundamental part of computation and data processing in IoT systems. This platforms provide functionalities from IoT Devices, enabling them to send data to the cloud. The cloud provides a guhe computing potential enabling user to deploy advanced analytics to the real time data provided by the IoT network. Furthermore, most recent development enable developer also to apply Machine Learning and AI capabilities to the system. Eventually these technologies will provide meaningful insights to users. Also in this area the market is expanding very fast, a part of the services will be presented at the end of this chapter.

3.4.5 Services

Services in IoT cover different areas. Starting from the simplest features, identity related services are the ones that identify the objects of the IoT network that will bring real word information to a virtual environment.

After identifying those objects, it is necessary to aggregate and standardize raw data. To do so, developers implement algorithms that collect data on a datawarehouse or cloud database.

Furthermore, when data are aggregated and standardized, they have to be used to take decisions, design improvement strategies and take actions. Smart systems allow also to implement closed-loop strategies where the system is programmed to react independently based on some rules. It is clear that AI and ML play an important role in this case but depending on the use case it is necessary to define carefully the constraints where the system can act independently or it need a human in the loop.

Eventually, all these services are aimed to be provided with continuity and in geographically spread areas.

Having this classification in mind it is interesting to review some IoT use cases. The application of IoT is wide and it implements innovative solution in different sectors like smart home, transport (self-driving vehicles, etc.), building automation, etc. In this paraph are presented some of the characteristics of use cases in industrial automation and smart grid.

Industrial automation is generally aimed to introduce and improve robotic presence in the production lines in order to increase productivity, efficiency and reduce human tasks. Generally, it enables machines and manufacturing appliances to produce complex products with a better control on processing, transportation, data collection and communication. IoT in this context or IIoT (Industrial IoT) is normally introduced as a technology to control and monitor operations, functionalities and productivity exchanging data through the internet. As mentioned in Chapter 3, IoT can be used to implement real time strategies or generate data to design improvement and predictive plans. Some examples related to the real-time capabilities can be related to the maintenance operations. Indeed, a connected machine that find a problem (ie. Current/temperature peak, abnormal vibrations, etc.) can send in real time a signal to the maintenance management service. Then, these data can be analyzed with ML algorithms in order to find a correlation between signals and start designing predictive maintenance strategies.

The same concept can be applied to quality control. Monitoring the physical data of a production machine and linking those data with quality control inspection can lead to the identification of critical phases in the production environment, thus design predictive quality processes. On the other hand, quality can be enhanced in real time connecting a smart camera equipped with automatic video analysis in order to help operators check the production quality during visual inspection tasks. Some interesting use cases related to these field can be found in automotive production where the production is complex and need to run at high production rates. Indeed, video analysis is useful where the products are complex and with a lot of specific part (i.e. flap positioning inspection, etc.). Another example is in the food production where the products may be transported at high speed on conveyors and need to be constantly checked by operators to guarantee specific requirements based on the visual aspect of the products.

Eventually, IIoT is useful also to connect information from enterprise system. Indeed, such systems like enterprise resource planning (ERP), manufacturing execution system (MES) and product lifecycle management (PLM) can benefit of crossing production data with IoT system. For instance, IoT enable the vertical integration of production and operation data with context data empowering operation managers to identify critical parameters and production shortfalls. A common KPI used to evaluate the overall performance is the Overall Equipment Efficiency (OEE). This indicator is based on the calculation of three parameters:

- Availability: defined as the ratio of total hours and the lost time
- Performance Rate: ratio of machine design speed and actual speed
- Quality Rate: ratio of good product and total products.

In smart grids, IoT is largely used to improve resources consumption. The most relevant example is related to energy consumption but also water management is being transformed with IoT, especially in cities where scarcity of water is a crucial aspect.

Deploying IoT in smart grids can connect several buildings with power and water network and give to suppliers key information to control and manage resources to improve the consumption and scale properly the network based on population variations. Furthermore, monitoring is the first step to forecast potential failures and improve services quality and the system resilience. Another topical issue related to smart grid and smart cities is also about pollution metering through air quality and water quality meters.

3.4.6 Semantics

This component of IoT systems is related to the possibility of getting useful information from different machines, sensors and systems in order to provide the required functionalities.
This step refers both the research of the methodology to individuate critical tasks and the modeling of the resources in order to effectively extracting value for the information collected.
Furthermore, it includes the utilization of the data from advanced data analysis methods. In order to do so, it is crucial that the system guarantee an horizontal integration between services and systems. It is therefore necessary to cover this aspect with common standards. Indeed, there are semantic web technologies to satisfy this requirement like Web Ontology Languade and Resource Description Framework. W3C that uses Efficient XML Interchange (EXI) as recommended format.

3.5 IoT Protocols and Standards

There are several protocols and standards in the IoT environment. These standards are generally provided to make it possible for developers to develop the required services. The organization that develop these protocols are several, the most famous are:

- World Wide Web consortium (W3c)
- Internet Engineering Task Force (IETF)
- IEEE
- EPCGlobal
- European Telecommunications Standards Institute (ETSI)

This chapter will investigate which are the most used protocols analyzing them and giving a perspective on which are the ones commonly used today and the ones expected to be adopted in the future.

3.5.1 Message Queue Telemetry Transport

Message queue Telemetry Transport (or MQTT) is an open message protocol that has been designed in order to deliver connection between devices, systems and machines. Indeed, the connection in this case utilizes a routing mechanism: one-to-one, one-to-many, many-to-many, making it an ideal candidate for machine-to-machine connection.

This system enables the transfer of messages between a server and the connected entities (i.e. sensors, phones, computer, etc.) relying on a publish-and-subscribe messaging paradigm (see Figure below).

In real applications, this protocol mainly addresses machine-to-machine and Internet of Things over low-bandwidth, high-latency and unreliable TCP/IP networks.

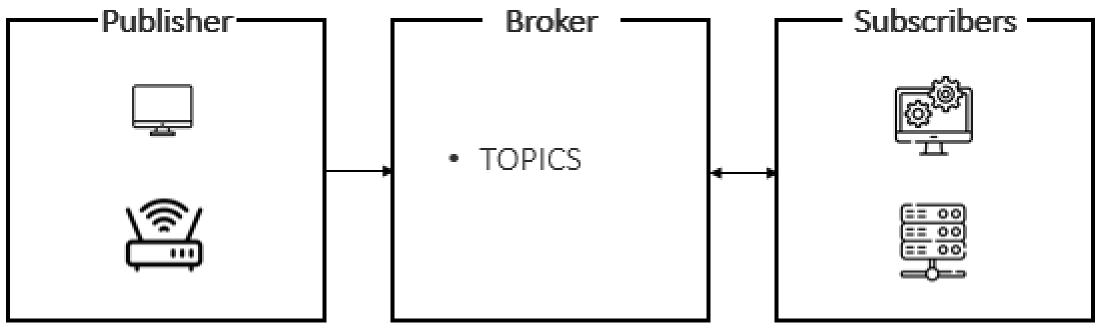


Figure 3.14 - MQTT

The subscribers represent the devices that would log in for specific topics in order to receive a signal from the broker when publisher publish some data of interest. After that, the publisher sends the data to the correct subscribers through the broker.

Moreover, the aim of the broker is also to control the authorization of the publishers and subscribers in order to guarantee the security requirements.

The adoption of this protocol is wide as it is lightweight, open and free. It has been approved by the Organization for the Advancement of Structured Information standards and it is also ISO standard.

For engineers and developers aiming to reduce designing embedded systems for uses cases with strict boundaries often consider using MQTT. Indeed, MQTT is an open source option that can be used to implement both on-edge and cloud services, offering developers different choices for the deployment of IoT systems.

It should also be considered that adopting an open source protocol reduce the dependency of the solution on the vendor. On the other hand, it does not provide support as private protocols and from a technical point of view it should be kept in mind that MQTT expose data in clear text. Thus, it is not the best option to deal with security.

3.5.2 Constrained Application Protocol

Constrained Application Protocol (CoAP) is a protocol developed especially as a web tranfer protocol and it used in constrained networks and nodes for machine-to-machine applications.

The communication model of Constrained Application Protocol is similar to the client-server model of HTTP, indeed the CoAP is based on Representational State Transfer (REST) that can be considered as a simpler way to exchange data between client and server than HTTP. Nevertheless, for machine-to-machine projects there is often an implementation of CoAP acting in both server and client roles.

Moreover, this protocol has some different functionalities from in respect of HTTP since to satisfy some IoT requirement like having noise and losses in the links and the need of low power consumption.

Even if this protocol has been developed on REST, it is possible to convert these two protocols very easily

In the figure below it is presented a general scheme of the CoAP protocol:

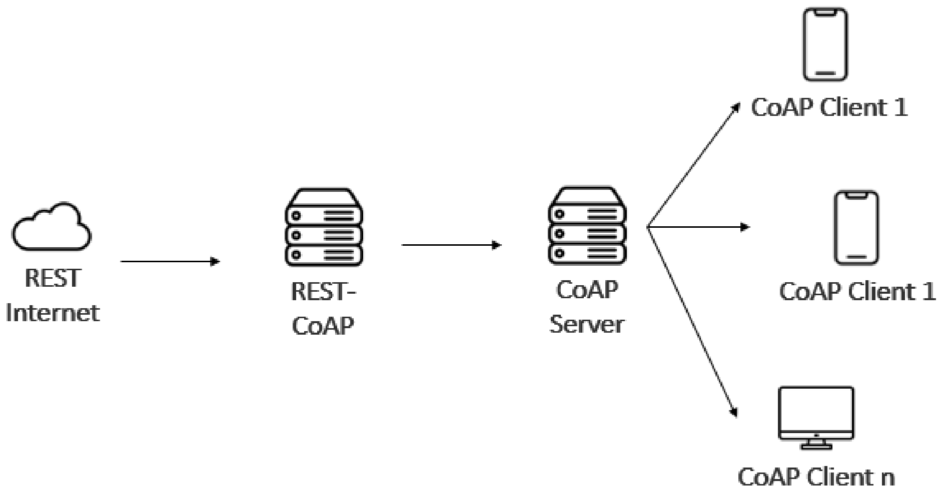


Figure 3.15 CoAP

The CoAP gained popularity in the open-sorce community with developments in C, C#, Java and Python. Nowadays, it is not widely adopted but the trend of IoT devices with small resources could accelerate the momentum of this protocol. It is important that developers check the security when using this protocol, especially when CoAP/HTTP mapping is a proxy. Furthermore, it is important to check that the IoT platform being used supports this protocol.

3.5.3 Advanced Message Queuing Protocol

The advanced Message Queuing Protocol (AMQP) is a standard developed in order to implement message-oriented middleware. This protocol supports message queuing and publish-and-subscribe messaging paradigm. AMQP enable several components or applications to communicate program-to-program. Defining a wire protocol by AMQP messages can be sent between software and message-oriented middleware are provided by several vendors.

Once the wire protocol is defined, AMQP systems are able to interoperate. The communication is managed by two main components showed in the figure below.

The frame in the figure below shows that the first four bytes show the frame size. The DOFF that stands for Data Offset, identify the position of the body inside the frame. The type gives the format and the aim of the frame.

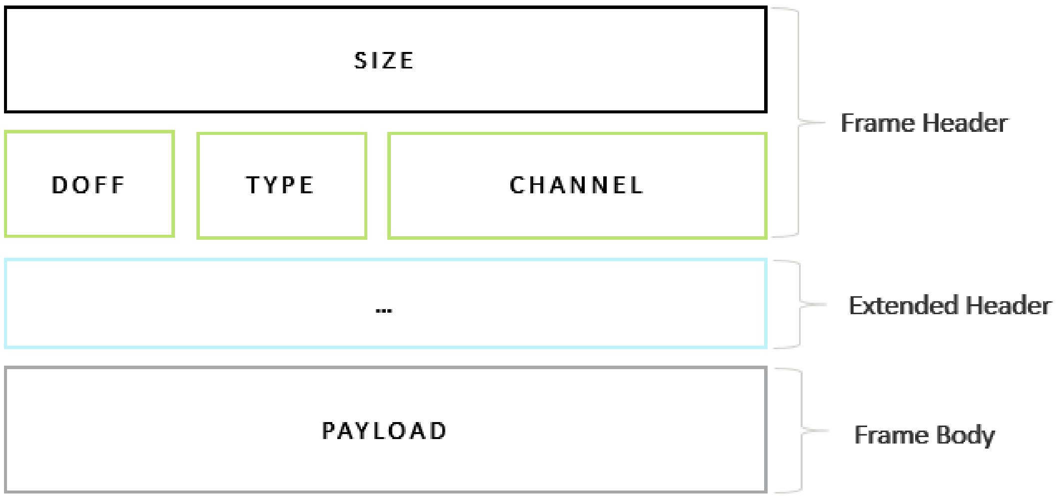


Figure 3.16 . AMQP

The choice of the adoption of AMQP depends mainly on the aim of the application. It is important to underline that interoperability of different versions of AMQP is not guaranteed. When developing a project on the short term this protocol should be take in consideration only if there is the need of integrating messages with other systems.

3.5.4 Data Distribution Services

Data Distribution Service (DDS) is different from the protocols introduced before as it has a broker-less architecture. This protocol uses multi-casting and that enhance reliability to its applications.

DDS is mostly used for for machine-to-machine communication, the integration with enterprise system and mobile devices.

The architecture of DDS consists of two layers:

- Data-Centric Publish-Subscribe: this layer aims to delver the information to subscribers
- Data-Local Reconstruction Layer: this layer is an interface to DPCS functionalities

The figure below shows the architecture of the DPCS:

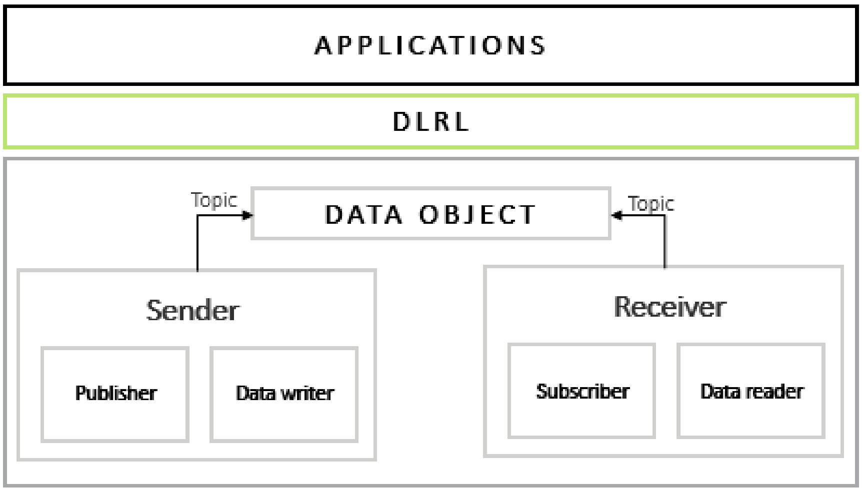


Figure 3.17 DDS

The architecture consists of five components:

- Publisher: this component shares the data into with other entities
- Data Writer: this component interact with the publisher in order to detect the values and data modifications
- Subscriber: it retrieves the data sent from the publisher and transfer them to the application
- Data Reader: it reads the data collected by the subscriber
- Topic: this is characterized by a name and a data type.

DDS first version came out in 2003, since then it evolved in order to cover real-time systems requirements. Nowadays, it has different implementation developed to design systems is several sectors: IIoT, energy, healthcare, communication, autonomous cars, etc. The DDS guarantee high performance in security and reliability and should be considered by developers when designing IoT systems with resourceful devices (automotive, autonomous vehicles, etc.)

3.5.5 Protocols Comparison

The protocol choice depends of different factors. In order to compare protocols performances, authors generally analyze the number of exchange messages between the client and the server in a determined timeframe.

It is interesting to compare the performance of MQTT and CoAP looking at the end to end transmission delay and bandwidth usage. CoAP result to deliver messages with higher delay then MQTT when the packet loss rate is low (D. Thangavel, et al. 2014). On the other hand, MQTT is worse when delivering small size messages, indeed CoAP overcome MQTT in creating less traffic.

Other research (N. De Caro, et al) studies performances of these two protocols in a mobile application and it turned out that MQTT has a bigger bandwidth usage and round trip time than CoAP. Referring to HTTP, this protocol result to be less efficient than CoAP in transmission time and energy usage.

Other protocols, like XMPP have been investigated to verify their applicability in real-time systems. For example, XMPP resulted to be a good choice if designing a real-time web application.

AMQP instead showed better results than REST webservices when exchanging a high volume of messages (G. Esposito, et al.). In analyzing DDS, this protocol scales well when the number of nodes is higher. Analysis on protocol performance is an interesting topic and it is useful for the first steps of IoT system design. By the way, further investigation based on the use case requirements and the environment are needed in order to select the correct protocol to connect all the system layers.

Here below is reported a table with the comparison of the main features of the protocols cited in this paragraph:

Protocols	**CoAP**	**MQTT**	**AMQP**	**DDS**	**HTTP**	**XMPP**
Transport	UDP	TCP	TCP	TCP	TCP	TCP
Publisher/Subscriber	Yes	Yes	Yes	Yes	Yes	Yes
Security	DTLS	SSL	SSL	SSL	SSL	SSL
QoS	Yes	Yes	Yes	Yes	Yes	No
Open-Source	Yes	Yes	Yes	Yes	Yes	Yes
Architecture	P2P	Broker	P2P Broker	Broker-less	P2P	P2P
Sponsor	IETF	OASIS	OASIS	OMG	IETF	IETF

Tab 3.1 – Protocol Comparison

3.6 IoT Cloud

Cloud implementation in manufacturing provide a solution to realize a cost effective, flexible and scalable implementation to companies.

Deploying cloud technologies in production environments allow organization to share resources and capabilities with innovative services that require less support and maintenance to the infrastructure.

Indeed, this technology allow to be highly versatile based on the resources needed from the system, allowing also to auto-scale virtual resources without any investment on hardware and infrastructure.

Furthermore, serverless architecture and ML models available on the cloud take advantage of cloud providers knowledge on computer science innovations, providing businesses advanced tools that can be considered as black box and only require to be tuned to develop meaningful use cases, cutting the development part for the customers.

It is clear that anytime a company decide to invest in this technology the trade off will be between internally develop such capabilities or invest in a recurrent cost to support cloud implementation and functionalities.

Therefore, before a company decide to carry out a cloud migration, the benefits of using such models has to be clear both from the side of users and providers.

The benefits are in good part generated when the system covers several manufacturing services.

Indeed, if the implementation does not aim to satisfy enough requirements for the customer it results to be tough to unleash the potential of such technology.

The link between cloud and IoT is clear since cloud implementation will be distinguished by a different architecture and also deploy IoT functionalities to the network.

Furthermore, the lower stack of manufacturing is related to the service generation by sensing and connection of different entities on the field. This is indeed the core value of IoT, the main feature of these systems has been presented in the previous chapters. After that the resources perceived has be to be organized and valuable services has to be disposed to the users.

This layer of sensing and feeding cloud system is a key aspect in the adoption of cloud systems. Indeed, customers can be concerned by sharing their manufacturing and knowledge on a network connected to the cloud. Accordingly, hugh standards in quality and security has be guaranteed in order to meet customer expectation and enable manufacturing plants to take advantage of this technology.

In Chapter 4.2 will be presented a use case where a hybrid cloud architecture has been deployed in production environment.

CHAPTER 4

Digital Innovation in Industry –Use Cases

4.1 IoT for energy carrier monitoring and optimization

Efficiency improvement is key for any organization to be competitive. Energy consumption is one of the main factors that needs to be controlled in order to reap resources and optimize processes.

Furthermore, the most recent regulation related to environmental and social sustainability set a challenging goal that needs to be met by companies, agencies, governments and institutions.

The first step that is needed to achieve a reduction in any field is to measure the parameter that needs to be decreased.

Moreover, modern companies use several energy carriers (electricity, HVAC, gas, compressed air, water, etc.) characterized by different unite of measures, time frames, costs. Therefore, it is necessary to project a system that is able to retrieve data from different sources, aggregate and standardize the data. IoT constitutes a tool to address this issue. Indeed the architecture presented in the previous chapter applies to this kind of scenario enabling companies to avail of the feature that follows:

- Wide spread connectivity to connect sensors and systems with several protocols
- Ingest data and store them in order to apply streaming analytics (analysis in real time) and historical analysis

- Contextualize field data with third party information (weather, building occupancy, shift, production line, etc.)
- Calculate the expected consumption of the entity being monitored in order to calculate efficiencies, losses and forecast the consumption needed. This feature is a major parameter when integrating renewable resources.
- Establish thresholds in order to react in real-time to contingency situation (energy spike, etc.)
- Give to plant managers a vision of the energy status with intuitive data visualization tools.

In the following chapter, will be presented a functional study of an energy optimization system based on IoT technology. Indeed, the functional building blocks of the system (energy sensors, functional layout, etc) will be presented and it will proposed an architecture and a dashboard wireframe to implement a web application that present the data to users.

4.1.1 Energy Sensors

As mentioned before, to design an IoT system it is mandatory to analyze the data sources. Both when working on green field or a brown field this part is crucial to define a proper solution.

Sometimes brown fields already provide valuable data that can be retrieved by the production line (i.e. PLC, SCADA, etc.) or enterprise systems that can help the definition of the use case and the collection of contextual information.

In any case, most of the times it is a good solution to identify non-intrusive sensors that can provide users reliable data with a high granularity without having to introduce major changes in the production line.

In this section are presented a list of non sensors that have been selected in order to collect data from energy carriers:

- Seneca T201 – Current Trasducer:

Figure 4.1 - Seneca T201

This an alternating current transducer that enables to convert the value of the measured current in a normalized industrial signal.

- Siemens Sentron PAC 2020

Figure 4.2 – Siemens Sentron

Siemens sentron is used to measure three-phase current, and power (apparent, active, reactive).

- EMU Pro 3/5 TCP/IP

Figure 4.3 – EMU Pro

This sensor also measure energy and power. It is therefore possible to see in real-time all data using a password at a protected website. It can be read remotely by via several protocols.

- Countis E4X

Figure 4.4 – Countis E4X

This sensor is also designed to measure three-phase current and can be connected to network via ethernet or remotely via Modbus.

- AEOTech Energy Meter

Figure 4.5 - AEOTech Energy Meter

This sensor is particularly used to record three-phase current through amperometric clamps. It is mostly used in smart-home applications.

- Qubino – Smart Meter

Figure 4.6 – Qubino Smart Meter

This is a Z-wave module to measure single-phase current. It measures current, power and power factor phi.

- Clamp-on ultrasonic flow meter for liquid

Figure 4.7 – Clamp-on sensor

This sensor is used to meter flow with bi-directional flow measurement. This is an example of non-intrusive sensor that can be applied in order to retrieve data without introducing any change to the environment.

- Energycam -EoL

Figure 4.8 - EnergyCam

This sensor provides OCR directly to analogic sensors to read the data and transfer it digitally in real-time.

4.1.2 Functional Layout

In order to study a functional solution implementing IoT for energy management, a simplified model of a plant is presented in this chapter. Plant layout can be very different depending on the process, the scheme presented in following figure below is intended as a example of the entities that could be monitored in a plant for polymers products:

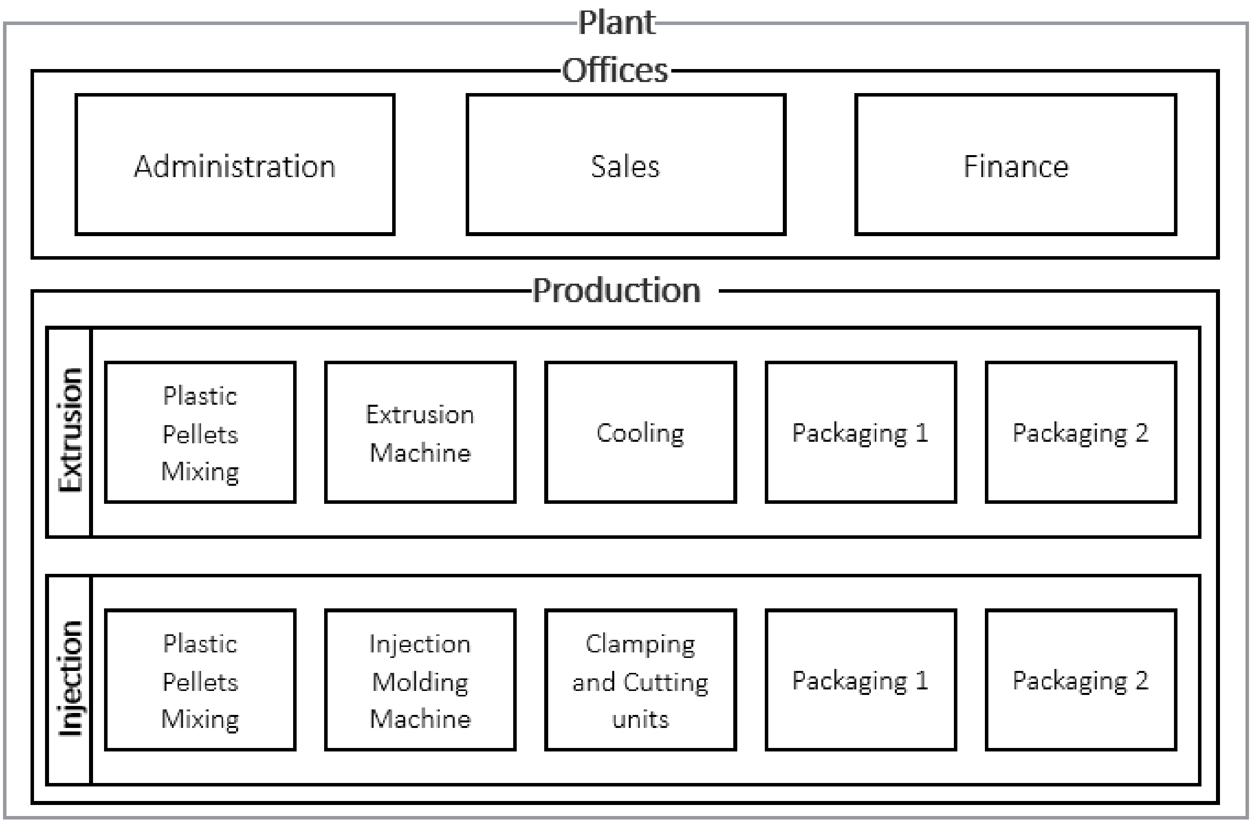


Figure 4.9 – Functional Layout

The lines represented in the scheme are two: one dedicated to the extrusion of material and the other for injection molding.

The lines consist of a station to provide raw plastic material, an extrusion and an injection molting machine, a cooling system, a station for clamping and cutting the molded products and two areas for packaging.

Energy carriers can be different based on the production, in this case it is considered that the most relevant are:

- Electricity
- Compressed air
- HVAC
- Natural Gas

In order to be compliant with standard reports the system will have to acknowledge the GRI (Global report Initiative) standards. Indeed, electricity and HVAC will be reported in kWh and natural gas and compressed air in SCFT (standard cubic foot). Even if HVAC is also powered by electricity, it normally makes sense to measure them separately since HVAC is high energy consuming component.

The data collected can also be connected to the real time cost of energy and it is therefore possible to provide to users also the information about how much money they are spending in real time and make analysis.

Assuming that in the examples the plant managers has already implemented a sensor network, the system aim will be to monitor consumption and losses for each part of the plant both directly - data directly retrieved from sensors - or indirectly – aggregating data.

4.1.3 Data and indirect data

In this scenario it is assumed that the following sensors will be installed:

Legenda:

- this icon indicates that the measurements is direct
- this icon indicates that the measurement is done via data aggregation

	Natural gas	Electricity	Compressed air	A/C
Plant				

Offices		🖥		🖥
Administration		((o))		((o))
Sales		((o))		((o))
Finance		((o))		((o))
Production	🖥	🖥	🖥	((o))
Extrusion	🖥	🖥	🖥	
Plastic Pellets Mixing		((o))		
Extrusion Machine	((o))	((o))		
Cooling		((o))		
Packaging 1		((o))	((o))	
Packaging 2		((o))	((o))	
Injection	🖥	🖥	🖥	
Plastic Pellet Mixing	((o))	((o))		
Clamping		((o))		
Packaging 1		((o))	((o))	
Packaging 2		((o))	((o))	

Tab 4.1 – Sensor Mapping

In Chapter 4.1.1 are presented some examples of the sensors that can be deployed.

4.1.4 Contextualization

Contextualizing information is a key step in order to improve knowledge deduction from data. Indeed, this represent one of the key values of IoT system that are able to connect from multiple sources on the field and aggregate data with sources on the web or from enterprise system. In this case the contexts that are considered to be relevant are:

- Shift: if the plant runs 24 hours the shift considered will be three:
    - 1: from 6 am to 2 pm
    - 2: from 2 pm to 10 pm
    - 3: from 10 pm to 6 pm
- Product: this data can be retrieved by the Manufacturing Execution System (MES)
- Production Order: collected by the Enterprise Resource Planning (ERP)
- Season: winter, summer, spring and autumn, can be get by the calendar
- Weather: sensor exposed to external environment or Web sources
- Volume: number of produced products

Contexts must be chosen strategically based on the desired analysis. For example it is interesting to cross the product or production order with global consumption data to identify which product consume more energy. Contextualization can also help users to decide which are the most critical components of the lines. This a crucial information to decide where to focus for analytics and to concentrate efforts to implement predictive solutions. Indeed, for predictive maintenance projects, anomalies in energy consumption are often considered as a signal of malfunctions and the patterns that can be identified through data can be used in order to make predictions.

4.1.5 Losses

In order to reduce energy consumption, it is necessary to measure it. Identifying losses in the energy grid is therefore important to identify where to act in the short term.

The type of losses that are relevant to be measured in this scenario are:

- Machine/Device performance: this kind of losses refer to the malfunctioning of a machine or device that is consuming more energy than expected. It is possible to set in the system a threshold of the value and compare to the actual consumption in order to set alarms and produce reports for the users.
- Unproductive losses: it can happen – especially in plant that are not highly automated – that operators leave a machine running even if it is not required. Or that employees leave light on also during the night.
- Transmission loss: this refers to the typical losses of energy in the transmission lines. A solution to directly measure transmission loss is to deploy a sensor at the start and at the end of the transmission line:

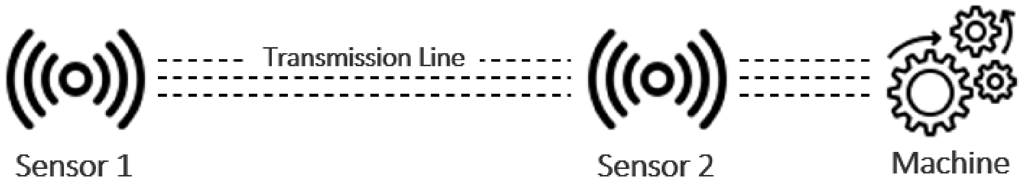


Figure 4.10 – Sensor Losses Scheme

This type of monitoring is directly connected to the consumption of electricity in the transmission line. Therefore, it is interesting that it has been successfully applied also in smart city projects in order to measure water losses in the waterworks, supplying the network with upstream and downstream sensors.

It important that losses are also monitored with a high granularity, but it is also a good practice to analyze where and at what timeframes they are more relevant. Indeed, the offices in plant will be open only during the shift 1 and 2, whilst during the third shift it is interesting to measure the unproductive losses.

4.1.6 Baseline

The baseline definition is an important feature to consider. Indeed, defining a baseline means providing users an expected value of the total amount of consumption of all energy carriers.

As a first step the baseline can be defined by reading the past energy bills or calculating the energy consumption.

Translating these values into a curve that evolves in time (based on season consumption, etc.) allow to compare it visually with the consumption values that will be presented in the dashboard. This will allow users to check if the energy consumption is overcoming the expect value of the system is efficient and is consuming less energy than expected.

4.1.7 Functional Architecture

In the figure below is reported a conceptual architecture.

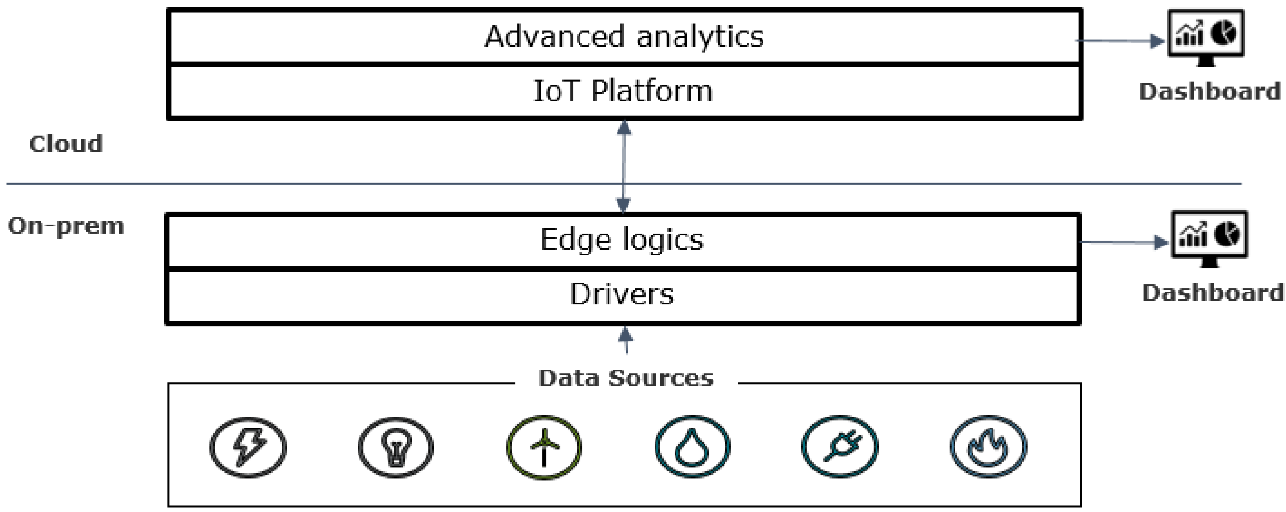


Figure 4.11 – Functional Architecture

Breaking down the functional requirement that have to be designed in order to

- Data collection: the data are retrieved on the field by the sensors
- Drivers: data can be transmitted in different protocols, drivers will collect and translate into a common format to dispose useful raw data
- Edge Logics: these logics perform pre-elaboration to the data and feed the edge dashboard for a visualization of real-time data and KPIs
- IoT Platform and Analytics: the pre-elaborated data are sent to the cloud to perform advanced analytics and create historical database
- Bidirectional communication: information elaborated from the cloud can be sent back to the edge stack in order to perform appliance on real time data
- Closed loop: on the edge part it is possible to set rules in order to implement real-time feedback to the system without sending data to the cloud.

4.1.8 Use case Benefits: Savings, regulation compliance and innovation

The study of this use case has been carried out to provide a functional guidance to carry out the development of such solutions.

The benefits of such solution can be seen on several levels: economical savings, environmental efficiency, regulation respect and digitalization enablement.

Indeed, the numbers projected for different sectors all indicates that investing in innovation for energy can lead to huge savings: the consultation group ‘‘Smart Manufacturing” foresees a saving potential of 10–40% at European level and highlights the importance of ICT (Information and Communication Technologies) as an enabler for energy efficiency (Gokan May, et al., 2016), at the Los Angeles Community College District, an audit team identified ten buildings to be integrated into the Energy Management System. The projected annual saving using the energy management system is $250,551 and 845,481 kWh (energy management system overview - sciencedirect), on smart lighting smart solid-state lighting in office buildings and industrial installations has the potential to reduce energy costs by 90 percent (Janessa Rivera, et al, Gartner) and the Silicon Valley-based company Enlighted, for example, claims to reduce clients' lighting bills by 60 to 70 percent and their air conditioning bills by 20 to 30 percent (*what are cost savings from industrial IoT - Technative*).

On the other hand, institutions are nowadays very attentive to the environmental impacts of companies and further regulations have to be respected by the companies in order to be compliant. For example, in Italy the legislative decree 254 of 30 December 2016 declares it is mandatory since 2017 for banks, assurances, companies with more than 500 employees or 20 million of balance-sheet or 40 million of turnover to declare a non-financial statement. This report must be audited as the financial one and must include information on environmental and social aspects: energy consumption, wastewater consumption, greenhouse gas emissions, etc. During the audit companies have to aggregate the data from several plant to demonstrate that the amount of consumes resources does not overcome specific limitation. This part can be time consuming and hard to carry out, using a digital system is a solution that can develop automated reports to be certified by regulators. Furthermore, implementing an IoT based energy management system allow users to develop innovative strategies on the energy management. An interesting development that can be carried out with such systems is related to the adaptive baseline generation. As mentioned, it is possible to define the baseline looking at bills at past consumption data. However, when the system is running it will create a structured database with exact records that can reach a high level of granularity. Those data can be used in order to develop a forecasting model of the baseline (or expected consumption curves). This is a key information for managers to calculate efficiencies and losses and also is an important support when evaluating the integration of renewable resources within the grid.

Eventually, the energy data can be used to model predictive maintenance models on machine. Indeed, the anomalies in the energy consumption can be considered an alarm bell for the malfunctions of machines and finding a correlation between these two parameters can lead to the development of predictive strategies that represent for companies a huge opportunity to optimize operation and costs.

4.2 Cloud solution for digital document classification

Engineers often must design complicated objects, such as specific tools and machine parts. The design process follows several steps and reviews, generating several digital and physical documents that are crucial components to design a product. To design these objects, nowadays engineers use computer-aided design (CAD) software. However, it also possible to find physical documentation that must be loaded into a digital archive. Indeed, many companies need to digitize a huge amount of documentation (photos, drawings, etc.) to integrate digital information.

Furthermore, engineering documentation (drawings, specifications, etc.) is an essential part of companies' know-how. Yet, a common issue for engineering teams is to keep the material structured and find it when needed, especially for companies that are conducting a reengineering of their processes and IT systems.

The time spent to reorganize documentation is often charged on the same people that are assigned to complex task, generating for companies a loss in terms of cost and employee's motivation towards their duties.

In order to help companies and organization solve these problems, cloud technology dispose a set of tools based on machine learning algorithms that can be deployed in order to automate the recognition of documents content and reorganize them.

This chapter will present an architecture deployed on Google Cloud Platform designed to solve such issue.

4.2.1 Document tagging – Cloud Architecture

This study will focus on a solution to deploy document tagging.

Document tagging refers to the process of assigning to a document a value that defines the features of it.

The figure reported below represent a proposed architecture that can be used to leverage machine learning capabilities from google cloud platform.

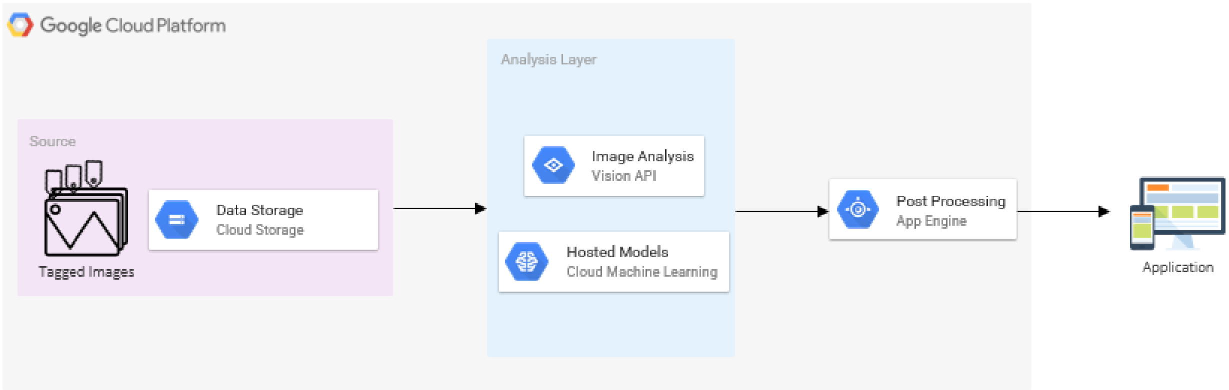


Figure 4.12 – Cloud Architecture

An architecture like the one presented in the figure above is simple to deploy in production. Indeed, all the components are based on GCP products that are scalable and serverless.

In order to train the machine learning algorithms developed by Google it is necessary to develop a set of tagged documents. The data with proper annotation can be stored on a cloud storage and be disposed to the machine learning engine.

The engine runs on Cloud Vision API and Cloud Machine learning whose details will be presented further in this chapter. The post processing data can be done via python solution and the tags will be stored in a interchange file format (i.e. JSON).

4.2.2 Data Labelling

If the dataset is not already labelled, developer can choose if using a data from a public database, labeling manually, or using a labelling tool.

Obviously, the choice depends on the final use of the system, if the label and the object are really specific of the process and are not common material to be found online, they will have to be labelled. The data labeling is often a service that is outsources by companies and organization that are conducting a project that involves machine learning. In any case, in order to define a data labeling framework user, have to at least define:

- Dataset: a collection of data that are representative of the samples that have to be analyzed. Data items can be updated in a Cloud storage bucket and it is a common practice to create a catalog of the images in a CSV file.
  For pictures it is a good practice to build a training model that consider at most a hundred times for the most common label that for the least common label.
- Label set: this task consists in listing all the possible labels of the considered dataset. As it will be presented in the next chapter Cloud Vision API is already trained to recognize objects and pictures characteristics, but it can not be so accurate with specific components. In this step, it is important that labels are easily distinguishable from one another and to define the lowest meanings overlap of labels. It is a good practice to define at least 20 labels in the label set.

4.2.3 Vision API and Auto ML Vision Object Detection

These are two technologies that are provided by GCP in order to make automated document analysis leveraging machine learning. Vision API consists of a pre-trained model that can be integrated in applications through REST and RPC APIs. The aim of such tool is to automatic assign labels to imagines and quickly classify them without the need of training a new model. On the other hand the model behind Vision API is static and as clearly it cannot be retrained. If the users must train a model that assign custom labels and categories from user-trained image model models, developers can use AutoML Vision Object Detection.

Indeed, this tool enables developers to train machine learning models to detect individual objects and features of pictures. It enables multiple object localization and provides information about where the object is in the picture.

In the table below there is a list of the main differences between these two tools:

Features	AutoML Vision	Vision API
REST/RPC APIs	x	x
Graphical UI	x	
Pre-trained Model		x
Classification with Custom labels	x	
Edge Deployment	x	
Object Detection (posistion, quantity)	x	x
Research enablement (similar objects in the dataset)		x
Automatic OCR		x
Popular logos and products identification		x
Attribute detection (dominant color, etc)		x

Tab 4.2 – AutoML – Vision API Comparison

4.2.4 App Engine

There are several ways to carry out the post processing (Jupyter notebook, python contenierized solution, etc.). Then in order to build application the it can be deployed the App Engine. This tool is an example of cloud computing platform as a service developed in order to support the development and hosting of webapplications in google.

Such solution allow application to run on multiple servers and enable them to automatic scale based on the application demand.

Such environment supports several language languages:

- Go
- PHP
- Java
- Python
- Node.js
- .NET
- Ruby
- Etc.

4.2.5 Use Case Benefits

Machine learning based on cloud is clearly unlocking valuable benefits to business and technical departments. Indeed, solution based on the technologies presented in this chapter meet many objectives of daily challenges like enhance the user experience, descrease operation costs and reduce errors.

Deploying these technologies can be useful both automation of documentation digitalization and consultation. Indeed, such modules can be integrated or develop a digital archive that makes use of machine learning to achieve functionalities like:

- Automatic Image Tagging
- Tagging on a specific dictionary based on users requirements

- Research of similar documents within the one analyzed by the app: this is one of the functionalities provided by Vision API
- Retraining of ML models based on users feedback:it is indeed possible to automatically track users feedback on the tags assigned by the algotirhm. Whenever a user change the model can be retrained in order to increase the system accuracy.

In this chapter it has been analyzed the possibility of applying such models by taking the one available on Google Cloud Platform. It is clear that this solutions enable a quick and reliable start-up process to introduce ML in the organization processes. By the way, users will have to consider the trade-off between deploying a “black-box” model like the one presented or developing – internally or in outsourcing – their own project, especially when the requirements of the users cannot be covered with cloud solutions.

CHAPTER 5

Conclusions

This thesis focused on three applications that demonstrate how digital solutions can provide new tools to solve several problems in companies, institutions and organizations. The digital innovation initiatives are very interesting to be studied and even if they are based on different principles, they share some common aspects, both on project management and technology.

Indeed, the steps for every organization in order to begin and conduct innovation projects always start with the identification of valuable knowledge from inside and outside to challenges and opportunities that can be solved through digital innovation. Furthermore, the development of such systems can be carried out developing new solutions, customizing an existing one. Once these systems are developed, they need to be installed and maintained, affecting technical departments and requiring changes in the organization management. This introduces the need of new professional figures, governance systems, training processes and procedures. A good practice in order to exploit such projects is to leverage existing knowledge, and existing systems for new purposes. Indeed, the internal resources are often underestimated and must be taken into account in order to successfully deliver the maximal value to business processes, products and services. Likewise, when these solutions are integrated within the environment they are applied into, their dynamics can in turn shape the organization processes. Furthermore, for information technology projects it is common to apply agile project management to check constantly if they meet user's requirements. Indeed, since IT products and services are based on cognitive processes, they need to be controlled in order to check that the expectations respect the business analysis and technical development, for this reason it is advisable to apply an adaptive methodology to ensure the best outcomes of such projects.

Regarding the NH90 virtual reality simulator, it presented good results in the flying behavior. Indeed, flying is at the first attempts difficult for people that is not trained, the trainees need to have some familiarity with the helicopter layout and controls. The training scenarios that have been implemented using the logic tools and scripts in VBS3 resulted to be a good starting point for the training, both multiplayer and single player, guaranteeing a good cross between the users' experience and the training process. The helicopter interiors have been modelled faithfully to the real one, in order to the build a reproduction of the flight dynamics and procedures as much as possible close to the real one when using virtual reality head mounted display. The motion platform has been installed and it resulted highly responsive to user's command. This enables a much higher level of immersivity during the simulation and reduces the effect of simulation sickness to the trainees. Indeed, having a real feedback from the commands decrease the discrepancy between the vestibular and the visual system.

The feedback collected with the automatic logging of user's response during the training phase generate a reliable data set. The data have been shared with psychologists that will analyze them in order to extract meaningful information related to user's attention and reaction to visual stimuli. This is an important aspect as the interaction between psychological aspects and virtual environment affects the effectiveness of the training in terms of reliability and trainee capabilities. Having a quantitative evaluation of the results by introducing human factors in the simulation is a key resource for scientists to identify shortfalls and enhance the systems performance. Indeed, modeling the human behavior represent an effective parameter to estimate the possible enhancement for the system and find correlations between the technological development and the real performances of it in training people. Furthermore, it has been possible to integrate the automation of data collection directly into the simulator software, reducing architecture complexity and saving time and costs. As future developments it would be interesting to study the integration of such simulator with free-roaming technology. Introducing free-roaming allow trainees to overcome the boundaries of spatial constraints (HMD wires connected to the computer) in the simulation environment enabling the scenarios to integrate complex procedures both on the flight phase and the terrain operations. Furthermore, this represents a point of contact between the immersive technologies and internet of things. Indeed, the sensors that need have to be deployed in order to realize a free-roaming environment are an example of IoT systems, that in this case is devoted to track body movement and transmit to the virtual avatar of the trainees in the simulation scenario. As mentioned, this would allow trainees to interact and follow faithfully real people movements in a virtual world.

About Internet of Things, enterprises and consumers will carry on asking for new IoT solutions and technologies on different vertical markets. For sure, the energy market will continue to be impacted by this technology seeing rises in the deployment of sensors and the application of machine learning and AI techniques to the collected by IoT infrastructures.

In the near feature the heterogenous environment of connectivity protocols will continue to exist. Companies and organization adopting IoT solution must adopt solution to standardize in real-time data sources in order to deduct meaningful information from the collected data. Developers can rely on market technologies or build plug-ins that automatically translate different protocol intro common and standardized data.

In the next few years instead of using modern connectivity methods like 5G for mobile, a wide range of wireless protocols will still be available that also overlay on similar IoT uses cases. Some of them also overlay in terms of IoT uses cases like Bluetooth, ZigBee or NFC. On the other hand, for long distance communication there are now new solutions related to low orbit satellites. An example of this technology already working is provided by the over 1400 Starlink satellites in orbit, also Amazon and China are investing in this field.

IoT applications are expected to rise in the healthcare sector by means of remote detection of patient condition remotely. The pandemic pushed this sector a lot since many people had to delay necessary care and did not have the chance to prevent their conditions, accelerating the need of developing good solutions for remote care. The infrastructure to develop such uses cases may not be different from the one presented in this work, but surely the healthcare experiences will have to be studied to find solution to get the right insights of patient’s conditions and provide application that are usable from a wide range of population.

On the industrial point of view also, IoT is continuously increasing in terms of adoption. Indeed, many users were reluctant to connect the machinery to IoT systems, but since many companies are switching to remote operations the need of a distribute connection became almost mandatory.

Monitoring machinery conditions enable several use cases. For example, the maintenance processes that require a frequent physical presence of specialized technicians, often are now being carried out in a blended way while connected machines can send notification only when the intervention is needed. Furthermore, monitoring the machine parameters enable to deploy downtime management strategies and, once the IoT system is running, design predictive strategies and as-a-service business models.

Moreover, realizing an intelligent perception and bringing granular information to a digital infrastructure is one of the bottlenecks for the implementation of cloud infrastructure. Indeed, without addressing this problem, users that have a limited visibility of field operations would lose many information and data support. On the other hand, connecting clean data with a cloud infrastructure enables innovative use cases whose capabilities can be acquired exploiting cloud modules developed by technology vendors. An example of this approach is presented in the last chapter of this work, related to the automatic recognition and tagging of technical documentation. These achievements would not have been possible without the ingestion of huge datasets to train such models. The approach proposed in this work made possible to analyze three main trends of digital innovation solutions and methodologies in relation with actual case studies. The simulator and the functional analysis of the systems presented enable the reader to get practical examples of digitalization projects. All the use cases have been validated in applied research context. Delivering such projects require the collaboration of cross disciplinary teams including subject matter experts, engineers, computer scientists and managers. Therefore, the author is active in continuing conducting further studies on these topics in order to face the challenges introduced by these innovations, discover new opportunities to apply extensively these capabilities to real world problems and to collaborate in different fields in order to foster creativity and develop innovative solutions.

Reference

Rajiv Kohli, Nigel P. Melville (2017) “Digital innovation: A review and synthesis”, Wiley

Agostino G. Bruzzone, Kirill Sinelshchikov, Marina Massei, Giuliano Fabbrini, Marco Gotelli, (2020) ”Extended Reality Technologies for Industrial Innovation”, 32nd European Modeling & Simulation Symposium, 17th International Multidisciplinary Modeling & Simulation Multiconference

Sepehr Alizadehsalehia, Ahmad Hadavib, Joseph Chuenhuei Huangc (2020) “From BIM to extended reality in AEC industry”, Automation in Construction

Jack Ratcliffe, Francesco Soave, Nick Bryan-Kinns, Laurissa Tokarchuck, Ildar Farkhatdinov (2021) “Extended Reality (XR) Remote Research: a Survey of Drawbacks and Opportunities”, Cap2.6, arXiv:2101.08046v1

Fei Tao; Ying Zuo; Li Da Xu; Lin Zhang “IoT-Based Intelligent Perception and Access of Manufacturing Resource”, IEEE Transactions on Industrial Informatics

Jianxin Wang, Ming K. Lim, Chao Wang d, Ming-Lang Tseng (2021) “The evolution of the Internet of Things (IoT) over the past 20 years”, Computers and Industrial Engineering

L. Minh Dang, Md. Jalil Piran, Dongil Han, Kyungbok Min and Hyeonjoon Moon (2019) “
A Survey on Internet of Things and Cloud Computing for Healthcare”, MDPI Electronics

Al-Fuqaha A., Guizani M. (2015)"Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications" , IEEE Communication Surveys and tutorials Vol. 17 NO.4

Peter Middleton, Kay Sharpington, Chad Eschinger, Alfonso Velosa, Peter Havart-Simkin, Jonathan Davenport, Emil Berthelsen, Chris Meering, Eric Goodness, Sandeep Unni, Bill Ray, Milly Xiang, Kevin Quinn, Alan Priestley (2021) "IoT Market View, 2020-2021", Gartner

By Michele Pelino, Paul Miller, Andre Kindness, Danny Mu, Julie Ask, Arielle Trzcinski with Merritt Maxim, Renee Taylor (2021) "Predictions 2021: Internet Of Things (IoT)", Forrester

D. Thangavel, X. Ma, A. Valera, H. Tan, and C. K. Tan, "Performance evaluation of MQTT and CoAP via a common middleware," in Proc.IEEE 9th Int. Conf. ISSNIP, 2014, pp. 1–6.

N. De Caro, W. Colitti, K. Steenhaut, G. Mangino, and G. Reali, "Comparison of two lightweight protocols for smartphone-based sensing," in Proc. IEEE 20th SCVT, 2013, pp. 1–6.

C. Esposito, S. Russo, and D. Di Crescenzo, (2008) "Performance assessment of OMG compliant data distribution middleware," in Proc. IEEE IPDPS,, pp. 1–8.

Kai Yan, Ying Cheng, Fei Tao (2016) "A trust evaluation model towards cloud manufacturing", The internation Journal of Advanced Manufacturing Technology

Lise Jaillant, Annalisa Caputo (2021) “Unlocking digital archives: cross-disciplinary perspectives on AI and born-digital data” – AI & Society

Agostino G. Bruzzone, Enrico Briano, Enrico Bocca, Marina Massei (2007) “Evaluation of the impact of different human factor models on industrial and business processes”, Simulation Modelling Practice and Theory

J. T. Reason and J. J. Brand (1975) “Motion Sickness”, London: Academic press

Gary E. Riccio and Thomas A. Stoffregen (1991) “An Ecological Theory of Motion Sickness Postural Instability”, Ecological Psychology, 3:195-240.

Cruz-Neira, C., Sandin, D.J., DeFanti, T.A., Kenyon, R.V., Hart, J.C. (1992) "The CAVE: audio visual experience automatic virtual environment", ACM Commun., 35 (6), 64–72

Berg, Vane (2017) “An Industry Case Study: Investigating Early Design Decision Making in Virtual Reality”, Computer inf, ASME

Hale, K. S., & Stanney, K. M. (2014). "Handbook of virtual environments", CRC Press, Boca Raton, FL

Hereld, M., Judson, I. R., & Stevens, R. L. (2000). "Introduction to building projection-based tiled display systems". IEEE Computer Graphics and Applications, 20. pp. 22-28

Cruz-Neira, C., Sandin, D. J., & DeFanti, T. A. (1993, September). Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In Proceedings of the 20th annual conference on Computer graphics and interactive techniques (pp. 135-142).

Cruz-Neira, C., Sandin, D.J., DeFanti, T.A., Kenyon, R.V., Hart, J.C. (1992) "The CAVE: audio visual experience automatic virtual environment", ACM Commun., 35 (6), 64–72.

Bruzzone, A. G., Longo, F., Agresta, M., Di Matteo, R., & Maglione, G. L. (2016b). Autonomous systems for operations in critical environments. In Proceedings of the Modeling and Simulation of Complexity in Intelligent, Adaptive and Autonomous Systems 2016 (MSCIAAS 2016) and Space Simulation for Planetary Space Exploration (SPACE 2016) (pp. 1-8).

VanderVliet, G. (1992). The Test, Evaluation, Development, and Use of a Manned Flight Simulator to Support Navy Developmental Testing of the V-22 Osprey (No. 921978). SAE Technical Paper.

Oberhauser, M., Dreyer, D., Mamessier, S., Convard, T., Bandow, D., & Hillebrand, A. (2015, August). Bridging the gap between desktop research and full flight simulators for human factors research. In International Conference on Engineering Psychology and Cognitive Ergonomics (pp. 460-471). Springer, Cham.

Mourtzis, D., Zogopoulos, V., & Vlachou, E. (2017). Augmented reality application to support remote maintenance as a service in the robotics industry. Procedia Cirp, 63(2017), 46-51.

Papachristos, N. M., Vrellis, I., & Mikropoulos, T. A. (2017, July). A comparison between oculus rift and a low-cost smartphone VR headset: immersive user experience and learning. In 2017 IEEE 17th International Conference on Advanced Learning Technologies (ICALT) (pp. 477-481). IEEE.

Elor, A., Powell, M., Mahmoodi, E., Hawthorne, N., Teodorescu, M., & Kurniawan, S. (2020). On shooting stars: Comparing cave and hmd immersive virtual reality exergaming for adults with mixed ability. ACM Transactions on Computing for Healthcare, 1(4), 1-22.

Amico, V., Bruzzone, A. G., & Guha, R. (2000, July). Critical issues in simulation. In SUMMER COMPUTER SIMULATION CONFERENCE (pp. 893-898). Society for Computer Simulation International; 1998.

Zeigler, B. P., & Lee, J. S. (1998, August). Theory of quantized systems: formal basis for DEVS/HLA distributed simulation environment. In Enabling Technology for Simulation Science II (Vol. 3369, pp. 49-58). International Society for Optics and Photonics.

Fujimoto, R. M. (2000). Parallel and distributed simulation systems (Vol. 300). New York: Wiley.

Liu, E. Theodoropoulos G.(2014)"Space-time matching algorithms for interest management in distributed virtual environments", ACM TOMACS, vol.24

Liu, R., Wang, L., Lei, J., Wang, Q., & Ren, Y. (2020). Effects of an immersive virtual reality-based classroom on students' learning performance in science lessons. British Journal of Educational Technology, e13434.

Graf, M., Timmerer, C., & Mueller, C. (2017, June). Towards bandwidth efficient adaptive streaming of omnidirectional video over http: Design, implementation, and evaluation. In Proceedings of the 8th ACM on Multimedia Systems Conference (pp. 261-271).

Steed, A., & Julier, S. (2013, March). Design and implementation of an immersive virtual reality system based on a smartphone platform. In 2013 IEEE Symposium on 3D User Interfaces (3DUI) (pp. 43-46). IEEE.

Ko, S. M., Chang, W. S., & Ji, Y. G. (2013). Usability principles for augmented reality applications in a smartphone environment. International journal of human-computer interaction, 29(8), 501-515.

Benford, S., & Giannachi, G. (2011). Performing mixed reality. The MIT Press.

Bruzzone, A. G., Longo F., Nicoletti L., Vetrano M., Bruno L., Chiurco A., Fusto C., Vignali G. (2016d). Augmented reality and mobile technologies for maintenance, security and operations in industrial facilities. 28th European Modeling and Simulation Symposium, EMSS 2016, pp. 355

Bruzzone, A. G., Massei, M., Maglione, G. L., Agresta, M., Franzinetti, G., & Padovano, A. (2016a). Virtual and augmented reality as enablers for improving the service on distributed assets. Proc. of I3M, Larnaca, Cyprus, September.

Mourtzis, D., Zogopoulos, V., & Vlachou, E. (2017). Augmented reality application to support remote maintenance as a service in the robotics industry. Procedia Cirp, 63(2017), 46-51.

Jacobson J.,Hwang Z.(2002) "Unreal tournament for immersive interactive theater", ACM Comm., 45 (1), 39–42

Hale, K. S., & Stanney, K. M. (2014). "Handbook of virtual environments", CRC Press, Boca Raton, FL

Kim, S. J., Jeong, Y., Park, S., Ryu, K., & Oh, G. (2018). A survey of drone use for entertainment and AVR (augmented and virtual reality). In Augmented Reality and Virtual Reality (pp. 339-352). Springer, Cham.

Muñoz-Saavedra, L., Miró-Amarante, L., & Domínguez-Morales, M. (2020). Augmented and Virtual Reality Evolution and Future Tendency. Applied Sciences, 10(1), 322.

Whisker, V. E., Baratta, A. J., Yerrapathruni, S., Messner, J. I., Shaw, T. S., Warren, M. E., Rotthoff E.S., Winters J.W., Clelland J.A. & Johnson, F. T. (2003). "Using immersive virtual environments to develop and visualize construction schedules for advanced nuclear power plants". In Proceedings of ICAPP Vol. 3, pp. 4-7.

Bruzzone, A. G., & Longo, F. (2013). 3D simulation as training tool in container terminals: The TRAINPORTS simulator. Journal of Manufacturing Systems, 32(1), 85-98.

Mastli, M., & Zhang, J. (2017). Interactive highway construction simulation using game engine and virtual reality for education and training purpose. In Computing in Civil Engineering 2017 (pp. 399-406).

Longo, F., Nicoletti, L., & Padovano, A. (2018). An interactive, interoperable and ubiquitous mixed reality application for a smart learning experience. International Journal of Simulation and Process Modelling, 13(6), 589-603.

Sagayam, K. M., & Hemanth, D. J. (2017). Hand posture and gesture recognition techniques for virtual reality applications: a survey. Virtual Reality, 21(2), 91-107.

Kumari, S., & Polke, N. (2018, August). Implementation Issues of Augmented Reality and Virtual Reality: A Survey. In International Conference on Intelligent Data Communication Technologies and Internet of Things (pp. 853-861). Springer, Cham.

Bowman, D. A., McMahan, R. P., & Ragan, E. D. (2012). Questioning naturalism in 3D user interfaces. Communications of the ACM, 55(9), 78-88.

Coelho, J. C., & Verbeek, F. J. (2014). Pointing task evaluation of leap motion controller in 3D virtual environment. Creating the difference, 78, 78-85.

Cruz-Neira, C., Sandin, D. J., & DeFanti, T. A. (1993, September). Surround-screen projection-based virtual reality: the design and implementation of the CAVE. In Proceedings of the 20th annual conference on Computer graphics and interactive techniques (pp. 135-142).

Muhanna, M. A. (2015) "Virtual reality and the CAVE: Taxonomy, interaction challenges and research directions", Journal of King Saud University-Computer and Information Sciences, 27(3), 344-361.

Jacobson J.,Hwang Z.(2002) "Unreal tournament for immersive interactive theater", ACM Comm., 45 (1), 39–42.

Advani, S., & Va, M. (1995). The design of a high-performance all-composite flight simulator motion platform. In Flight Simulation Technologies Conference (p. 3382).

Chen, J., Fang, Y. C., Loftin, R. B., Leiss, E. L., Lin, C. Y., & Su, S. (2001). An immersive virtual environment training system on real-time motion platform. Proc. of the Computer Aided Design and Computer Graphics, 2, 951-954.

Thöndel, E. (2010). Electric motion platform for use in simulation technology–design and optimal control of a linear electromechanical actuator. In Proceedings of the World Congress on Engineering and Computer Science (Vol. 2).

Lee, S. M., Xia, K., & Son, H. (2020). Robust Tracking Control of Spherical Motion Platform for Virtual Reality. IEEE Transactions on Industrial Electronics.

VanderVliet, G. (1992). The Test, Evaluation, Development, and Use of a Manned Flight Simulator to Support Navy Developmental Testing of the V-22 Osprey (No. 921978). SAE Technical Paper.

Pollini, L., Innocenti, M., & Petrone, A. (2008). Novel motion platform for flight simulators using an anthropomorphic robot. Journal of Aerospace Computing, Information, and Communication, 5(7), 175-196.

Schroeder, J. A. (1999). Helicopter flight simulation motion platform requirements, NASA Technical Report, Ames Research Center, Moffett Field, California.

Hell, S., & Argyriou, V. (2018, December). Machine learning architectures to predict motion sickness using a virtual reality rollercoaster simulation tool. In 2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR) (pp. 153-156). IEEE.

Amico, V., Bruzzone, A. G., & Guha, R. (2000, July). Critical issues in simulation. In SUMMER COMPUTER SIMULATION CONFERENCE (pp. 893-898). Society for Computer Simulation International.

Chang, C. C., & Tsai, W. H. (2001). Vision-based tracking and interpretation of human leg movement for virtual reality applications. IEEE transactions on circuits and systems for video technology, 11(1), 9-24.

Cameron, C. R., DiValentin, L. W., Manaktala, R., McElhaney, A. C., Nostrand, C. H., Quinlan, O. J., ... & Gerling, G. J. (2011, April). Hand tracking and visualization in a virtual reality simulation. In 2011 IEEE systems and information engineering design symposium (pp. 127-132). IEEE.

Shin, J., An, G., Park, J. S., Baek, S. J., & Lee, K. (2016). Application of precise indoor position tracking to immersive virtual reality with translational movement support. Multimedia tools and applications, 75(20), 12331-12350.

Park, J., Chen, J., & Cho, Y. K. (2017). Self-corrective knowledge-based hybrid tracking system using BIM and multimodal sensors. Advanced Engineering Informatics, 32, 126-138.

Zhang, Y., Fei, Y., Xu, L., & Sun, G. (2015, July). Micro-IMU-based motion tracking system for virtual training. In 2015 34th Chinese Control Conference (CCC) (pp. 7753-7758). IEEE.

Angelino, C. V., Baraniello, V. R., & Cicala, L. (2012, July). UAV position and attitude estimation using IMU, GNSS and camera. In 2012 15th International Conference on Information Fusion (pp. 735-742). IEEE.

Wang, L. J., Hu, Q. N., & Shu, H. (2011). Research on Technologies of Machining Virtual Experiment. Donghua Daxue Xuebao(Ziran Ban), 37(4), 462-466.

Foxlin, E., Harrington, M., & Pfeifer, G. (1998, July). Constellation: a wide-range wireless motion-tracking system for augmented reality and virtual set applications. In Proceedings of the 25th annual conference on Computer graphics and interactive techniques (pp. 371-378).

Pence, H. E. (2010). Smartphones, smart objects, and augmented reality. The Reference Librarian, 52(1-2), 136-145.

Pimentel, K., Teixeira (1992) "Virtual reality: through the New Looking Glass", TAB Books, NYC

George L. Danek (1993) “Vertical Motion Simulator Familiarization Guide”, NASA Technical Memorandum, Mofflet Field, California 94035-1000.

Kennedy, R.S., Lane, N.E., Berbaum, K.S., & Lilienthal, M.G(1993) “Simulator Sickness Questionnaire: An enhanced method for quantifying simulator sickness, International Journal of Aviation Psychology”, 3:203-220.

Fabiano, B., Parentini, I., Ferraiolo, A., Pastorino (1995) “A century of accidents in the Italian industry, Safety Science”, 21: 65-74

Giribone P., Bruzzone A.G. & Tenti M.(1996) "Local Area Service System (LASS): Simulation Based Power Plant Service Engineering & Management", Proc. of XIII Simulators International Conference, New Orleans LA, April 8-11

Minister of transport Canada (1996) Helicopter flight training manual.

Brown, L. (1999). “A Radar History of World War II”. Institute of Physics Publishing, Bristol

Djellal, F., & Gallouj, F. (1999) “Services and the search for relevant innovation indicators: a review of national and international surveys”, Science and Public Policy, 26(4), pp. 218-232

Bowman, D., Hodges, L. (1999) "Formalizing the design, evaluation and application of interaction techniques for immersive virtual environments", J. Visual Lang. Comput., 37–53

Amico Vince, Bruzzone A.G., Guha R. (2000) "Critical Issues in Simulation", Proceedings of Summer Computer Simulation Conference, Vancouver, July

Montgomery D.C. (2000) "Design and Analysis of Experiments", John Wiley & Sons, New York

Bass, T. (2000) "Intrusion detection systems and multisensor data fusion" Communications of the ACM, 43(4), pp.99-105

Hereld, M., Judson, I. R., & Stevens, R. L. (2000). "Introduction to building projection-based tiled display systems". IEEE Computer Graphics and Applications, 20. pp. 22-28.

Nakatsu R., Tosa, N.(2000)"Active Immersion: the goal of Communications with interactive agents", Proc. of Int.Conf. on Knowledge-Based Intelligent Engine

Joseph J., LaViola Jr., (2000), “A Discussion of Cybersickness in Virtual Environments”, Department of Computer Science, Brown University.

Marco Gotelli, Giuliano Fabbrini, Agostino Bruzzone (2017) “Calibration of simulation models and VR/AR training and maintenance”, Master Thesis, Università degli Studi di Genova

Mobley, R. K. (2001) “Plant engineer's handbook”, Butterworth-Heinemann, Oxford, UK

Majumdar, A., Polak, J. (2001). “Estimating capacity of Europe's airspace using a simulation model of air traffic controller workload”. Journal of the Transportation Research Board, (1744), 30-43.

Richards A., J. Bellingham, M. Tillerson, and J. P. (2002) “How: Co-ordination and control of multiple UAVs”, Proc. of the AIAA Guidance, Navigation, and Control Conference, Monterey, CA, August

Feddema, J.T.; Lewis, C.; Schoenwald, D.A., (2002) "Decentralized control of cooperative robotic vehicles: theory and application, "Robotics and Automation, IEEE Transactions on, vol.18, no.5, pp.852,864, Oct

Jacobson J.,Hwang Z.(2002) "Unreal tournament for immersive interactive theater", ACM Comm., 45 (1), 39–42

Sherman, W. R., & Craig, A. B. (2003) “Understanding Virtual Reality—Interface, Application, and Design. Presence”, Morgan Kaufmann Publisher, SF, 12(4)

Whisker, V. E., Baratta, A. J., Yerrapathruni, S., Messner, J. I., Shaw, T. S., Warren, M. E., Rotthoff E.S., Winters J.W., Clelland J.A. & Johnson, F. T. (2003). "Using immersive virtual environments to develop and visualize construction schedules for advanced nuclear power plants". In Proceedings of ICAPP Vol. 3, pp. 4-7

Keating, C., Rogers, R., Unal, R., Dryer, D., Sousa-Poza, A., Safford, R., Peterson W. & Rabadi, G. (2003) “System of systems engineering”, Engineering Management Journal, 15(3), 36-45

Vail D. & M. Veloso, (2003) “Dynamic multi-robot coordination”, Multi-Robot Systems: From Swarms to Intelligent Automata, Vol II, pp. 87-100.

Stilwell D. J., A. S. Gadre, C. A. Sylvester and C. J. Cannell (2004) “Design elements of a small low-cost autonomous underwater vehicle for field experiments in multi-vehicle coordination”, Proc. of the IEEE/OES Autonomous Underwater Vehicles, June, pp. 1-6

National Research Council. (2004) “Assessing the national streamflow information program”, National Academies Press, Washington DC, USA

Stilwell D. J., A. S. Gadre, C. A. Sylvester and C. J. Cannell (2004) “Design elements of a small low-cost autonomous underwater vehicle for field experiments in multi-vehicle coordination”, Proc. of the IEEE/OES Autonomous Underwater Vehicles, June, pp. 1-6

Jones, D. (2005) “Power line inspection-a UAV concept”, Proc. of the IEE Forum on Autonomous Systems, Ref. No. 11271, November

Keating, C. B., Sousa-Poza, A., & Kovacic, S. (2005) “Complex system transformation: a system of systems engineering (SoSE) perspective”, Proc. of 26th ASEM National Conference, pp. 200-207

Keegan, D. (2005) “The incorporation of mobile learning into mainstream education and training”, Proc. of World Conference on Mobile Learning, Cape Town, October.
Shah, A. P., et al (2005, June). “Analyzing air traffic management systems using agent based modeling and simulation” In Proceedings of the 6th USA/Europe Seminar on Air Traffic Management Research and Development

Bruzzone, A. G. et al. (2006) “Simulation and Optimization as Decision Support System in Relation to Life Cycle Cost of New Aircraft Carriers”, Proceedings of Modelling Simulation and Optimization, Gaborone, Botswana

Ross, S., D. Jacques, M. Pachter, and J. Raquet, (2006) “A Close Formation Flight Test for Automated Air Refueling,” Proceedings of ION GNSS-2006, Fort Worth, TX, Sep

Grocholsky, B., Keller, J., Kumar, V., Pappas, G., (2006) "Cooperative air and ground surveillance", Robotics & Automation Magazine, IEEE, vol.13, no.3, Septmber, pp.16-25

Jans, W., Nissen, I., Gerdes, F., Sangfelt, E., Solberg, C. E., & van Walree, P. (2006) “UUV covert acoustic communications- preliminary results of the first sea experiment”, in Techniques and technologies for unmanned autonomous underwater vehicles- a dual use view”, RTO Workshop SCI-182/RWS-016, Eckernförde, Germany

Ross, S., D. Jacques, M. Pachter, and J. Raquet, (2006) “A Close Formation Flight Test for Automated Air Refueling,” Proceedings of ION GNSS-2006, Fort Worth, TX, Sep

Doherty, P., & Rudol, P. (2007) "A UAV search and rescue scenario with human body detection and geolocalization", Proceedings of the Australian Conference on Artificial Intelligence, Vol. 4830, December, pp. 1-13

Tanner H. G. (2007) “Switched UAV-UGV cooperation scheme for target detection”, IEEE International Conference on Robotics and Automation, Roma, Italy, April, pp. 3457-3462

Tanner H.G., D.K. Christodoulakis, (2007) “Decentralized cooperative control of heterogeneous vehicle groups”, Robotics and Autonomous Systems 55,pp 811–823

Tanner H. G. (2007a) “Switched UAV-UGV cooperation scheme for target detection”, IEEE International Conference on Robotics and Automation, Roma, Italy, April, pp. 3457-3462.

Tanner H.G., D.K. Christodoulakis, (2007b) “Decentralized cooperative control of heterogeneous vehicle groups”, Robotics and Autonomous Systems 55,pp 811–823

Tumer, K., Agogino, A. (2007, May). “Distributed agent-based air traffic flow management” In Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems (p. 255). ACM

Shafer, A.J., Benjamin, M.R., Leonard, J.J., Curcio, J., (2008) "Autonomous cooperation of heterogeneous platforms for sea-based search tasks", Oceans, , September 15-18, pp. 1-10

Bruzzone A.G., E. Bocca (2008) "Introducing Pooling by using Artificial Intelligence supported by Simulation", Proc.of SCSC2008, Edinbourgh, UK

Jamshidi, M. (2008) "Introduction to system of systems. System of Systems Engineering", CRC Press, NY, pp. 1-43

Mittal, S., Zeigler, B. P., Martín, J. L. R., Sahin, F., & Jamshidi, M. (2008) “Modeling and Simulation for Systems of Systems Engineering”, in Systems of Systems Innovation for 21th Century, Wiley & Sons, NYC

Sousa-Poza A., Kovacic S., Keating C. (2008) "SoSE: An Emerging Multidiscipline", Int.Journal of Systems Engineering, Vol.1, Nos.1/2

Rhodes, D. H., Valerdi, R., & Roedler, G. J. (2009) “Systems engineering leading indicators for assessing program and technical effectiveness”, Systems Engineering, 12(1), 21-35

Stroeve, S. H., Blom, H. A., Bakker, G. B. (2009). “Systemic accident risk assessment in air traffic by Monte Carlo simulation”, Safety science, 47(2), 238-249.

Sujit, P. B., Sousa, J., Pereira, F.L., (2009) "UAV and AUVs coordination for ocean exploration", Oceans - EUROPE, vol., no., pp.1,7, 11-14 May

Tether, T. (2009) "Darpa Strategic Plan", Technical Report DARPA, May

Sujit, P. B., Sousa, J., Pereira, F.L., (2009) "UAV and AUVs coordination for ocean exploration", Oceans - EUROPE, vol., no., pp.1,7, 11-14 May

Tether, T. (2009) "Darpa Strategic Plan", Technical Report DARPA, May

Prof. Alessandro Colonna ,(2009), “Costruzione di strade ferrovie e aeroporti”, University of Bari, Bari (Italy).

Nolan, M., (2010), “Fundamentals of air traffic control”, Delmar, Boston, MA

Mark McCall, Bob Mury, May 2010, Distributed Interactive Simulation, DIS PDG Chair.

Jamshidi, M. (2011) “System of systems engineering”, Innovations for the twenty-first century, vol 58, John Wiley & Sons, NYC

Ncube, C. (2011) “On the Engineering of Systems of Systems: key challenges for the requirements engineering community”, Proc. of IEEE Workshop on Requirements Engineering for Systems, Services and Systems-of-Systems (RESS), August, pp. 70-73

Merabti, M., Kennedy, M., & Hurst, W. (2011) “Critical infrastructure protection: A 21 st century challenge”, Proc. of IEEE Int.Conf. on Communications and Information Technology, ICCIT, March, pp. 1-6

Bürkle, A., Segor, F., Kollmann, M. (2011). "Towards autonomous micro uav swarms". Journal of intelligent & robotic systems, 61(1-4), pp. 339-353

Cárdenas, A. A., Amin, S., Lin, Z. S., Huang, Y. L., Huang, C. Y., & Sastry, S. (2011) "Attacks against process control systems: risk assessment, detection, and response", Proceedings of the 6th ACM Symposium on Information, Computer and Communications Security, March, pp.355-366

Merabti, M., Kennedy, M., & Hurst, W. (2011) “Critical infrastructure protection: A 21 st century challenge”, Proc. of IEEE Int.Conf. on Communications and Information Technology, ICCIT, March, pp. 1-6

Bruzzone A.G., Fadda P, Fancello G., Massei M., Bocca E., Tremori A., Tarone F., D'Errico G. (2011a) "Logistics node simulator as an enabler for supply chain development: innovative portainer simulator as the assessment tool for human factors in port cranes", Simulation October 2011, vol. 87 no. 10, p. 857-874, ISSN: 857-874

Bruzzone A., Massei M., Longo F., Madeo F., (2011b). Modeling and simulation as support for decisions makers in petrochemical logistics. Proceedings of the 2011 Summer Computer Simulation Conference, pp. 130

Dai, F. (Ed.). (2012) “Virtual reality for industrial applications”, Springer Science & Business Media

Quero, S., Botella, C., Pérez-Ara, M. A., Navarro, M., Baños, R. M., Maciá, M. L., & Rodríguez, E. (2012). The use of Augmented Reality for safety in health: The European Project ANGELS. In ICERI2012 Proceedings (pp. 215-218). IATED

Francesca Giardini, Frèdèric Amblard, (2012), “Multi-Agent-Based Simulation”, Springer, Spain.

Kastek, M., Dulski, R., Zyczkowski, M., Szustakowski, M., Trzaskawka, P., Ciurapinski, W., Grelowska G., Gloza I., Milewski S, Listewnik, K. (2012) “Multisensor system for the protection of critical infrastructure of seaport” In Proc. of SPIE, Vol. 8288, May

Shkurti, F., Anqi Xu, Meghjani, M., Gamboa Higuera, J.C., Girdhar, Y., Giguere, P., Dey, B.B., Li, J., Kalmbach, A., Prahacs, C., Turgeon, K., Rekleitis, I., Dudek, G., (2012)"Multi-domain monitoring of marine environments using a heterogeneous robot team", Proc. of IEEE Intelligent Robots and Systems (IROS), vol., no., pp.1747,1753, October 7-12

Spillane, J. P., Oyedele, L. O., & Von Meding, J. (2012) "Confined site construction: An empirical analysis of factors impacting health and safety management", Journal of Engineering, Design and Technology, 10(3), pp.397-420

Bruzzone, A.G., Berni, A., Fontaine, J.G., Cignoni, A., Massei, M., Tremori, A., Dallorto, M., Ferrando, A. (2013c) “Virtual Framework for Testing/Experiencing Potential of Collaborative Autonomous Systems”, Proc. of I/ITSEC, Orlando. FL USA

Magrassi C. (2013) “Education and Training: Delivering Cost Effective Readiness for Tomorrow's Operations“,ITEC Keynote Speech, Rome, May

Maravall D., J. de Lopea,b, R. Domíngueza, (2013) “Coordination of communication in robot teams by reinforcement learning”, Robotics and Autonomous Systems 61, pp.661–666

Nano, G., & Derudi, M. (2013) "A critical analysis of techniques for the reconstruction of workers accidents", Chemical Engineering, 31

Biocca, F., & Levy, M. R. (2013) "Communication in the Age of Virtual Reality", Routledge, London

Pérez-Ara, M. A., Quero, S., Navarro, M. V., Botella, C., & Baños, R. M. (2013) “Augmented Reality for Safety at Work: Needs Analysis of the Priority Risks for Safety”, Proc. of Health Context in Spain, INTED, pp. 812

Seidel, R. J., & Chatelier, P. R. (Eds.). (2013). "Virtual reality, training’s future? Perspectives on virtual reality and related emerging technologies". Springer Science & Business Media.

Nystrom Robert , (2014), Decoupling Patterns, Game Programming Patterns, 5:215

Pizzella, L. A. E. (2014) “Contributions to the Configuration of Fleets of Robots for Precision Agriculture”, Thesis, Universidad Complutense, Madrid, Spain, May

Siebert, S., & Teizer, J. (2014) “Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system”, Automation in Construction, 41, pp.1-14

Valavanis, K. P., & Vachtsevanos, G. J. (2014) "Handbook of unmanned aerial vehicles", Springer Publishing Company, NYC

Bruzzone A.G., Massei M., Agresta M., Poggi S., Camponeschi F., Camponesch M. (2014) “Addressing Strategic Challenges on Mega Cities through MS2G”, Proceedings of MAS, Bordeaux, France, September 12-14

Clarke, R., & Moses, L. B. (2014) “The Regulation of Civilian Drones' Impacts on Public Safety” Computer Law & Security Review, 30(3), 263-285

Pizzella, L. A. E. (2014) “Contributions to the Configuration of Fleets of Robots for Precision Agriculture”, Thesis, Universidad Complutense, Madrid, Spain, May

Valavanis, K. P., & Vachtsevanos, G. J. (2014) "Handbook of unmanned aerial vehicles", Springer Publishing Company, NYC

Siebert, S., & Teizer, J. (2014) “Mobile 3D mapping for surveying earthwork projects using an Unmanned Aerial Vehicle (UAV) system”, Automation in Construction, 41, pp.1-14

Perlman, A., Sacks, R., & Barak, R. (2014). "Hazard recognition and risk perception in construction". Safety science, 64, pp. 22-31.

Liu E., Theodoropoulos G.(2014)"Space-time matching algorithms for interest management in distributed virtual environments”, ACM TOMACS, vol.24

Zhang, J., Hou, H. T., & Chang, K. E. (2014) “UARE: Using reality-virtually-reality (RVR) models to construct Ubiquitous AR environment for e-Learning context”, Proc. of IEEE Science and Information Conference (SAI), August, pp. 1007-1010

Alzahrani, A., Callaghan, V., & Gardner, M. (2014) “Towards the Physical Instantiation of Virtual People and Components in Physical Mixed-Reality Tele-Presence Environments”, Proc. of Intelligent Environments Workshops, pp. 285-294, July

Benes, F., & Kodym, O. (2014) “Application of Augmented Reality in Mining Industry”, Proc. 14th SGEM GeoConference on Informatics, Geoinformatics and Remote Sensing, Vol. 1, 1-35, June 19-25

Hale, K. S., & Stanney, K. M. (2014). "Handbook of virtual environments", CRC Press, Boca Raton, FL

Kenyon, A., Van Rosendale, J., Fulcomer, S., & Laidlaw, D. (2014). "The design of a retinal resolution fully immersive VR display", Virtual Reality (VR), IEEE, pp. 89-90
Bruzzone, A. G., Massei, M., Tremori, A., Longo, F., Nicoletti, L., Poggi, S., Bartolucci C., Picco E. & Poggio, G. (2014b) “MS2G: simulation as a service for data mining and crowd sourcing in vulnerability Reduction” Proc. of WAMS, Istanbul, September.

Bruzzone, A. G., Massei, M., Tremori, A., Poggi, S., Nicoletti, L., & Baisini, C. (2014a) “Simulation as enabling technologies for agile thinking: training and education aids for decision makers” International Journal of Simulation and Process Modelling 9, 9(1-2), 113-127

Dorn A.W. (2014) “Aerial Surveillance: Eyes in the Sky” in Air Power in UN Operations: Wings for Peace (A. Walter Dorn, Ed.), Ashgate Publishing, Farnham, UK, pp. 119-134

Harvey C., Stanton N.A., (2014). Safety in System-of-Systems: Ten key challenges, Safety Science, vol. 70, pp. 358-366

Longo F., Chiurco A., Musmanno R., Nicoletti L., (2015). Operative and procedural cooperative training in marine ports. Journal of Computational Science, vol. 10, pp. 97-107.

Kehoe, B., Patil, S., Abbeel, P., & Goldberg, K. (2015) "A survey of research on cloud robotics and automation", IEEE Transactions on automation science and engineering, 12(2), pp.398-409

Kim, D. H., Kwon, S. W., Jung, S. W., Park, S., Park, J. W., & Seo, J. W. (2015) "A Study on Generation of 3D Model and Mesh Image of Excavation Work using UAV", Proceedings of the International Symposium on Automation and Robotics in Construction, Vol. 32, Vilnius, January

Leão, D. T., Santos, M. B. G., Mello, M. C. A., & Morais, S. F. A. (2015) "Consideration of occupational risks in construction confined spaces in a brewery", Occupational Safety & Hygiene III, 343

Merwaday, A., & Guvenc, I. (2015) "UAV assisted heterogeneous networks for public safety communications", Proc. of IEEE Wireless Communications and Networking Conference Workshops, March, pp. 329-334

Apvrille, L., Roudier, Y., & Tanzi, T. J. (2015) "Autonomous drones for disasters management: Safety and security verifications", Proc. 1st IEEE URSI Atlantic, May

Floreano, D., & Wood, R. J. (2015) “Science, technology and the future of small autonomous drones”, Nature, 521(7553), 460

Kehoe, B., Patil, S., Abbeel, P., & Goldberg, K. (2015) "A survey of research on cloud robotics and automation", IEEE Transactions on automation science and engineering, 12(2), pp.398-409

Kim, D. H., Kwon, S. W., Jung, S. W., Park, S., Park, J. W., & Seo, J. W. (2015) "A Study on Generation of 3D Model and Mesh Image of Excavation Work using UAV", Proceedings of the International Symposium on Automation and Robotics in Construction, Vol. 32, Vilnius, January

Kovacevic, M. S., Gavin, K., Oslakovic, I. S., & Bacic, M. (2016). "A new methodology for assessment of railway infrastructure condition". Transportation research procedia 14, pp. 1930-1939

Leão, D. T., Santos, M. B. G., Mello, M. C. A., & Morais, S. F. A. (2015) "Consideration of occupational risks in construction confined spaces in a brewery", Occupational Safety & Hygiene III, 343

Merwaday, A., & Guvenc, I. (2015) "UAV assisted heterogeneous networks for public safety communications", Proc. of IEEE Wireless Communications and Networking Conference Workshops, March, pp. 329-334

Bednarz, T., James, C., Widzyk-Capehart, E., Caris, C., & Alem, L. (2015). "Distributed collaborative immersive virtual reality framework for the mining industry" in Machine Vision and Mechatronics in Practice. Springer Berlin Heidelberg. pp. 39-48

Chafkin M. (2015) "Why Facebook's $2 Billion Bet on Oculus Rift might one day connect Everyone on Earth", Vanity Fair Hive, October

Le, Q. T., Pedro, A. K. E. E. M., Lim, C. R., Park, H. T., Park, C. S., & Kim, H. K. (2015) "A framework for using mobile based virtual reality and augmented reality for experiential construction safety education", International Journal of Engineering Education, 31(3), 713-725

Muhanna, M. A. (2015) "Virtual reality and the CAVE: Taxonomy, interaction challenges and research directions", Journal of King Saud University-Computer and Information Sciences, 27(3), 344-361

Safir, I. J., Shrewsberry, A. B., Issa, I. M., Ogan, K., Ritenour, C. W., Sullivan, J., & Issa, M. M. (2015). Impact of remote monitoring and supervision on resident training using new ACGME milestone criteria. Can J Urol, 22, 7959-7964

Documentation Team (2016) GIMP- GNU Image Manipulation Program (User Manual).

Bower, M., Lee, M. J., & Dalgarno, B. (2016) “Collaborative learning across physical and virtual worlds: Factors supporting and constraining learners in a blended reality environment”, British Journal of Educational Technology

Bruzzone A., Longo F., Nicoletti L., Vetrano M., Bruno L., Chiurco A., Fusto C., Vignali G. (2016a). Augmented reality and mobile technologies for maintenance, security and operations in industrial facilities. 28th European Modeling and Simulation Symposium, EMSS 2016, pp. 355.

Bruzzone A.G., Massei M., Maglione G., Agresta M., Franzinetti G., Padovano A. (2016b) “Virtual and Augmented Reality as Enablers for Improving the Service on Distributed Assets”, Proc. of I3M, Larnaca, Cyprus, September

Bruzzone A.G., Massei M., Maglione G.L., Di Matteo R., Franzinetti G. (2016c) "Simulation of Manned & Autonomous Systems for Critical Infrastructure Protection", Proc. of DHSS, Larnaca, Cypurs, September

Gonzalez, D. S., Moro, A. D., Quintero, C., & Sarmiento, W. J. (2016, August). Fear levels in virtual environments, an approach to detection and experimental user stimuli sensation”, Proc. of XXI IEEE Symposium on Signal Processing, Images and Artificial Vision (STSIVA), pp. 1-6

Lindgren, R., Tscholl, M., Wang, S., & Johnson, E. (2016) "Enhancing learning and engagement through embodied interaction within a mixed reality simulation". Computers & Education, 95, 174-187

Peña-Rios, A., Hagras, H., Gardner, M., & Owusu, G. (2016) “A Fuzzy Logic based system for Mixed Reality assistance of remote workforce”, Proc. IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), July, pp. 408-415

Scudellari M. (2016) "Google Glass Gets a Second Life in the ER", IEEE Specturm, May 25

Spanu S., Bertolini M., Bottani E., Vignali G., Di Donato L., Ferraro A., Longo F., (2016). Feasibility study of an augmented reality application to enhance the operators' safety in the usage of a fruit extractor. International Food Operations and Processing Simulation Workshop, FoodOPS 2016, pp. 70.

Wagner K. (2016) "Two Years Later: Facebook's Oculus Acquisition Has Changed Virtual Reality Forever", Recode, March 24

Bruzzone A.G., Maglione G.L. (2016) "Complex Systems & Engineering Approaches", Simulation Team Technical Report, Genoa

Davis M., Proctor M., Shageer B., (2016). A Systems-Of-Systems Conceptual Model and Live Virtual Constructive Simulation Framework for Improved Nuclear Disaster Emergency Preparedness, Response, and Mitigation. Journal of Homeland Security and Emergency Management, vol. 13, no. 3, pp. 367-394

Altawy, R., & Youssef, A. M. (2016) "Security, Privacy, and Safety Aspects of Civilian Drones: A Survey" ACM Transactions on Cyber-Physical Systems, 1(2), 7

Bruzzone A.G., Longo F., Massei M., Nicoletti L., Agresta M., Di Matteo R., Maglione G.L., Murino G., Antonio Padovano A. (2016a) “Disasters and Emergency Management in Chemical and Industrial Plants: Drones simulation for education & training”, Proc. of MESAS, Rome, June 15-16

Bruzzone A.G., Massei M., Longo F., Cayirci E., di Bella P., Maglione G.L., Di Matteo R. (2016b) “Simulation Models for Hybrid Warfare and Population Simulation”, Proc. of NATO Symposium on Ready for the Predictable, Prepared for the Unexpected, M&S for Collective Defence in Hybrid Environments and Hybrid Conflicts, Bucharest, Romania, October 17-21

Bruzzone A.G., Massei M., Maglione G.L., Di Matteo R., Franzinetti G. (2016c) “Simulation of Manned & Autonomous Systems for Critical Infrastructure Protection”, Proc. of I3M, Larnaca, Cyprus, September

Pulina, G., Canalis, C., Manni, C., Casula, A., Carta, L. A., & Camarda, I. (2016) “Using a GIS technology to plan an agroforestry sustainable system in Sardinia”, Journal of Agricultural Engineering, 47(s1), 23-23

Sanchez-Lopez, J. L., Pestana, J., de la Puente, P., & Campoy, P. (2016) "A reliable open-source system architecture for the fast designing and prototyping of autonomous multi-uav systems: Simulation and experimentation", Journal of Intelligent & Robotic Systems, 84(1-4), pp.779-797

Spanu S., M. Bertolini, E. Bottani, G. Vignali, L. Di Donato, A. Ferraro, F. Longo (2016) "Feasibility study of an Augmented Reality application to enhance the operators’ safety in the usage of a fruit extractor", Proc. FoodOPS, Larnaca, Cyprus, September 26-28

Steven M. La Valle, (2017), Virtual Reality, University of Illinois Cambridge University

VBS3 Manuals, (2017), VBS3 17.2.1, Bohemia Interactive Simulations, Vltavská 3101/24, Smíchov, 150 00 Prague 5, Czech Republic.

Alexander Ripkens, Thomas Alexander, DeLFI, (2017), Lecture Notes in Informatics (LNI), Konzeption eines Systems zur netzwerkgestützten Ausbildung mit Mixed Reality Technologien, Fraunhofer FKIE, Human Factors, Zanderstraße. 5, 53177 Bonn, Deutschland.

McCurry Justin (2017) "Dying robots and failing hope: Fukushima clean-up falters six years after Tsunami", The Guardian, March 9

Salvini, P. (2017) “Urban robotics: Towards responsible innovations for our cities”, Robotics and Autonomous Systems, Elsevier

Di Donato (2017) “Intelligent Systems for Safety of Industrial Operators, the Role of Machines & Equipment Laboratories”, SISOM Workshop, Rome

Palazzi, E., Caviglione, C., Reverberi, A.P., Fabiano, B. (2017) “A short-cut analytical model of hydrocarbon pool fire of different geometries, with enhanced view factor evaluation”, Process Safety and Environmental Protection, August

Salvini, P. (2017) “Urban robotics: Towards responsible innovations for our cities”, Robotics and Autonomous Systems, Elsevier

Gonzalez-Franco, M., Pizarro, R., Cermeron, J., Li, K., Thorn, J., Hutabarat, W. & Bermell-Garcia, P. (2017)
“Immersive Mixed reality for Manufacturing Training” Frontiers, 4(3), 1

Palazzi, E., Caviglione, C., Reverberi, A.P., Fabiano, B. (2017) "A short-cut analytical model of hydrocarbon pool fire of different geometries, with enhanced view factor evaluation”, Process Safety and Environmental Protection, August

Tatic, D., & Tešic, B. (2017) “The application of augmented reality technologies for the improvement of occupational safety in an industrial environment”, Computers in Industry, 85, 1-1

Bruzzone A.G., et al. (2017) "A STRATEGIC SERIOUS GAME ADDRESSING SYSTEM OF SYSTEMS ENGINEERING", Proc. of I3M, Barcelona, September

Bruzzone A.G., Massei M., Agresta M., di Matteo R., Sinelshchikov K., Longo F., et al. 2017, "AUTONOMOUS SYSTEMS & SAFETY ISSUES: THE ROADMAP TO ENABLE NEW ADVANCES IN INDUSTRIAL APPLICATIONS", Proc. of MAS, Barcelona, September

BRUNNER Elektronik AG, Factsheet 6DOF MONTION 100, Industriestrasse 27,CH - 8335 Hittnau

Dr. W. (Wim) Bles, Dr. J.E (Bernd) de Graaf, Dr. M. (Mark) Wentink, Msc, Desdemona, Deence, Security and Safety. Kampweg 3769 ZG Soesterberg

TM 55-1526-237 Series, TM 55-2840- Series, UH-60 BLACK HAWK FLIGTH SIMULATOR

Unity on-line Manual.

Sitography

https://www.oculus.com/.
https://www.vive.com/us/.
http://www.flightlink.com/base-page/advanced-rotor-wing-package/.
http://www.acme-worldwide.com/products/dynamic-motion-seats/helicopters/nh-90-dms/.
http://www.motion-simulators.com/.
https://bisimulations.com/products/virtual-battlespace.
https://library.vuforia.com/content/vuforia-library/en/articles/Training/ObjectRecognition.html
https://library.vuforia.com/articles/Training/Image-Target-Guide.