The Application of Mixed Reality Within Civil Nuclear Manufacturing and Operational Environments

EngD Thesis

Author:
Stephen Reddish
s.reddish@namrc.co.uk

27th February 2019

Abstract

This thesis documents the design and application of Mixed Reality (MR) within a nuclear manufacturing cell through the creation of a Digitally Assisted Assembly Cell (DAAC). The DAAC is a proof of concept system, combining full body tracking within a room sized environment and bi-directional feedback mechanism to allow communication between users within the Virtual Environment (VE) and a manufacturing cell. This allows for training, remote assistance, delivery of work instructions, and data capture within a manufacturing cell.

The research underpinning the DAAC encompasses four main areas; the nuclear industry, Virtual Reality (VR) and MR technology, MR within manufacturing, and finally the 4^{th} Industrial Revolution (IR4.0). Using an array of Kinect sensors, the DAAC was designed to capture user movements within a real manufacturing cell, which can be transferred in real time to a VE, creating a digital twin of the real cell. Users can interact with each other via digital assets and laser pointers projected into the cell, accompanied by a built-in Voice over Internet Protocol (VoIP) system. This allows for the capture of implicit knowledge from operators within the real manufacturing cell, as well as transfer of that knowledge to future operators. Additionally, users can connect to the VE from anywhere in the world. In this way, experts are able to communicate with the users in the real manufacturing cell and assist with their training. The human tracking data fills an identified gap in the IR4.0 network of Cyber Physical System (CPS), and could allow for future optimisations within manufacturing systems, Material Resource Planning (MRP) and Enterprise Resource Planning (ERP).

This project is a demonstration of how MR could prove valuable within nuclear manufacture. The DAAC is designed to be low cost. It is hoped this will allow for its use by groups who have traditionally been priced out of MR technology. This could help Small to Medium Enterprises (SMEs) close the double digital divide between themselves and larger global corporations. For larger corporations it offers the benefit of being low cost, and, is consequently, easier to roll out across the value chain. Skills developed in one area can also be transferred to others across the internet, as users from one manufacturing cell can watch and communicate with those in another. However, as a proof of concept, the DAAC is at Technology Readiness Level (TRL) five or six and, prior to its wider application, further testing is required to asses and improve the technology.

The work was patented in both the UK (S. Reddish et al., 2017a), the US (S. Reddish et al., 2017b) and China (S. Reddish et al., 2017c). The patents are owned by Rolls-Royce and cover the methods of bi-directional feedback from which users can interact from the digital to the real and vice versa.

Stephen Reddish

Mixed Mode Realities in Nuclear Manufacturing

Key words: Mixed Mode Reality, Virtual Reality, Augmented Reality, Nuclear, Manufacture, Digital Twin, Cyber Physical System

Contents

List of Figures

List of Tables

Glossary

List of Acronyms

Notation	Description
ABWR	Advanced Boiling Water Reactor
AGR	Advanced Gas Reactor
ALARA	As Low As Reasonably Acceptable
ALARP	As Low As Reasonably Possible
AMR	Advanced Modular Reactor
AMRC	Advanced Manufacturing Research Centre
API	Application Programming Interface
AR	Augmented Reality
ASCII	American Standard Code for Information Interchange
ASME	American Society of Mechanical Engineers
AWS	Amazon Web Service
B2B	Business to Business
BGS	British Geological Survey
BIM	Building Information Modelling
BOOM	Binocular Omni-Orientated Monitor
CAAP	Computer Assisted Assembly Planning
CAD	Computer Aided Design
CAVE	Computer Assisted Virtual Environment
CFD	Computational Fluid Dynamics

Notation	Description
CG	Computer Generated
CGN	China General Nuclear
CNNC	China National Nuclear Corporation
CPPS	Cyber Physical Production System
CPS	Cyber Physical System
CRT	Cathode Ray Tube
DAA	Digitally Assisted Assembly
DAAC	Digitally Assisted Assembly Cell
DAC	Design Augmented by Computer
DBEIS	Department for Business, Energy & Industrial Strategy
DES	Discrete Event Simulation
DoD	Department of Defence
DOE	Department of Energy
DoF	Degrees of Freedom
DPI	Dye Penetrant Inspection
DRMSE	Distributed Real Manufacturing Simulation Environment
DSE	Distributed Simulation Environment
EAC	Environmentally Assisted Cracking
EDF	Energy De France
EngD	Engineering Doctorate
EoL	End of Life
EPA	Environmental Protection Agency
EPR	Evolutionary Power Reactor
ERP	Enterprise Resource Planning
FEA	Finite Element Analysis

Notation	Description
FOSS	Free and Open Source Software
FoV	Field of View
FT	Financial Times
GDA	Generic Design Assessments
GNS	General Nuclear System Limited
GPS	Global Positioning Satellite
GUI	Graphical User Interface
HAL	Hardware Abstraction Layer
HCI	Human Computer Interaction
HMD	Head Mounted Display
HS&E	Health Safety & Environment
HUD	Head Up Display
ICS	Instrument & Control Systems
ICT	Information & Communications Technology
IoF	Incredibility of failure
IoT	Internet of Things
IP	Internet Protocol address
IR4.0	4^{th} Industrial Revolution
IVR	Immersive Virtual Reality
JET	Joint European Torus
JIT	Just In Time
JSF	Joint Strike Fighter
JSON	JavaScript Object Notation
KAERI	Korea Atomic Energy Research Institute
LEEP	Large Expanse Enhanced Perspective

Notation	Description
LH	Left Hand
LoD	Level of Detail
MES	Manufacturing Executions System
MMO	Massively Multi-player Online game
MoU	Memorandum of Understanding
MR	Mixed Reality
MRP	Material Resource Planning
MSR	Molten Salt Reactor
NDE	Non-Destructive Evaluation
NDT	Non-Destructive Testing
NIA	Nuclear Industry Association
NNL	National Nuclear Laboratory
NPP	Nuclear Power Plant
NPS	National Policy Statement
OEE	Overall Equipment Effectiveness
ONR	Office of Nuclear Regulation
PDM	Product Data Management
PDMS	Product Data Management System
PhD	Doctorate of Philosophy
PI	Place Illusion
PLM	Product Life-cycle Management
PP&C	Production Planning & Control
PPE	Personal Protective Equipment
PsI	Plausibility Illusion
PTSD	Post Traumatic Stress Disorder

Notation	Description
PWR	Pressurised Water Reactor
QA	Quality Assurance
RDP	Remote Desktop Protocol
RE	Research Engineer
RFID	Radio-Frequency Identification
RH	Right Hand
ROS	Robot Operating System
RR	Rolls-Royce
SDK	Software Development Kit
SME	Small to Medium Enterprise
SMR	Small Modular Reactor
ToF	Time of Flight
TRL	Technology Readiness Level
UKAEA	United Kingdom Atomic Energy Authority
UNC	University of North Carolina
VAPP	Virtual Assembly Process Planning
VAT	Virtual Assembly Training
VCASS	Coupled Airbourne Systems Simulator
VE	Virtual Environment
VIEW	Virtual Interface Environment Workstation
VIVED	Virtual Visual Environment Display
VM	Virtual Manufacturing
VoIP	Voice over Internet Protocol
VR	Virtual Reality
VRTK	Virtual Reality Toolkit

Notation	Description
WNA	World Nuclear Association

CHAPTER 1

Introduction

This thesis combines research from three main areas. The UK nuclear industry, UK Manufacturing, and Mixed Reality (MR). The culmination of the research is the creation of the Digitally Assisted Assembly Cell (DAAC). The DAAC is a mixed reality manufacturing cell, that seeks to address a number of gaps identified within the fields of research.

The UK nuclear industry is now post *'nuclear renaissance'* and has started construction on its first nuclear reactor in 30 years. The gap in manufacturing has led to a significant skills gap, and the retirement of most of the nuclear manufacturing capabilities within the country. The need to preserve knowledge of the ageing workforce, while recruiting and training a new workforce has never been greater within the industry.

Nuclear reactors require the highest level of safety and therefore regulatory code. This is clearly defined by the Office of Nuclear Regulation (ONR) (ONR, 2016) and conforms to the principles that risks should be kept As Low As Reasonably Possible (ALARP). The requirements of high conformance to meet the regulatory codes can lead to an increase in cost of the manufacturing processes. This is in part due to the levels of testing required to create enough statistical confidence in manufacturing methods, and testing. As the safety requirements for reactors has increased, so has the complexity of the reactors from generation to generation. Although this leads to a high component count, the repetition of parts is still low. This is because of the high energy density of nuclear fuel, and the move to ever increasing size of commercial reactors. The larger gen 3+ reactors proposed for the UK produce energy at a greater economic advantage with regards to fuel burn density vs replenishment, this conversely leads to the requirement of fewer reactors. In this way nuclear components can be considered both *High Value* and *Low Volume*.

Chapters 2, 3, 4, can be considered overviews of; the nuclear industry and manufacture, Mixed Reality and Virtual Reality (VR), and 4^{th} Industrial Revolution (IR4.0) respectively. Any readers well versed in any of these areas should feel welcome to skip to the final sections of the chapters which reflect upon their importance with regards to the DAAC. The chapters are included, as the work undertaken combines three nominally disparate areas of research.

Chapter 2 starts with a broad background of the nuclear industry within the UK, then focuses on reactor designs, and the potential for new reactors within the UK. A summary of potential challenges and opportunities derived from this research is then presented. The following sections then detail the challenges faced by the nuclear manufacturing industry. These include issues surrounding the supply of high value low volume components while meeting tight conditions of supply and regulation. Details are then presented of a few unique challenges associated with manufacturing for the nuclear industry including the need for highly trained staff who can accommodate rapid process change. The wide spread use of specialist equipment for both handling and manufacturing components within an industry that requires zero manufacturing defects. The chapter concludes with a summary of the challenges associated with manufacturing within the nuclear industry.

The use of MR within manufacturing has been pioneered by the motoring industry since training simulators were first proposed for driving schools. More recent advances in mobile phone technology has led to a resurgence of interest in VR as a technology and the creation of a companies producing light weight high resolution VR or Augmented Reality (AR) ready headsets. This has allowed users to interact with an entirely computer generated environment at a one to one scale. When coupled with the widespread use of Computer Aided Design (CAD) and simulation software the ability for users to interact with their data has reached an unprecedented level.

Chapter 3 starts with a definition of MR and where both VR and AR fit within the broader spectrum of MRs. The chapter then expands upon two concepts key to this thesis, VR and AR. Aspects of MR such as immersion and haptics are discussed in the following section, with special attention paid to visual cues such as depth indicators and methods of stereoscopy. The paradigm of Data, Information, Knowledge and Wisdom is discussed at length, and leads into methods of data capture and information within manufacturing. The challenges that surround the use of these data sets within a Virtual Environment (VE) are discussed with the use of past pioneering use of VR. The chapter concludes talking about computer simulation as a means of converting data into information, and more specifically: Discrete Event Simulation (DES), Continuous, and Monte Carlo simulation.

A review of the literature covering the academic work within the field of MR is contained within Chapter 4. The first sections cover key definitions within the field such as: VE, virtual presence, sensory feedback and interactivity. A larger section follows this covering the history of MR, from Huxleys early simulators, to modern Head Mounted Display (HMD) based systems. The first few sections conclude by identifying four main parties taking part in MR research as: the military, university research departments, VR enthusiasts, and MR technology based start ups. The literature review then continues by reviewing those papers associated with VR in manufacturing and in combination with DES, assembly process planning, and workforce training. A discussion section then sets out both challenges and benefits of using MR in three main areas of manufacture: design, manufacture, and in-service. The review concludes with a reflection on how MR can be seen as a way of structuring information from real manufacturing systems. As the industrial collection of data increases, technologies such as AR and VR can play an important role in converting the data into information, and knowledge. Industrial examples of how MR is used to tackle challenges posed in Chapter 3 are discussed before the chapter ends asking where the advances made in other industrial and technological centres could be applied within nuclear manufacturing.

Chapter 5 continues the focus on industry, covering what is known as the IR4.0, with sections on: Cyber Physical System (CPS), Digital Twins, and the Smart Factory. The opening sections provide a brief overview of what the IR4.0 is and the implications that this may have for manufacturing companies. The principal concepts of data, connectivity and analytics are discussed, and the benefits of horizontal and vertical data integration for manufacturing systems are made. The chapter then raises the challenges posed by standardisation, IT infrastructure & security, and Qualifications within the workforce, associated with implementing concepts of IR4.0. The chapter then focuses on CPSs and the benefits and challenges of creating self configuring systems with the ability to self optimise in real time. Storage of the large data sets created by CPS is discussed and the concept of the Digital Twin is introduced. Potential data sources and the storage of context relevant data and information at a product level is covered. The chapters penultimate section is on the use of digital twins within simulations and as analytical tools that can aid decision making processes. The chapter concludes by stating that:

- The IR4.0 presents ideal use cases for MR, as a means of visualising context-sensitive data.
- The IR4.0 adds requirements for more advanced training methods for the workforce to allow them to keep pace with technological advances.
- CPS can be used to disseminate technology both horizontally along the supply chain, and vertically within a company.
- That Digital Twins offer effective means of preserving knowledge in an industry undergoing rapid change.

In Chapter 6 a Rolls-Royce project which centred around the use of data sensors within a manufacturing cell is used as an example of data integration. The Research Engineer (RE) was tasked with displaying context-sensitive data, polled from a sensor suite, within a VE. The chapter details the implementation of this with the use of the Unity3D game engine. Starting with an overview of the project and its aims, before laying out the data flow from external sensors to within the game engine. A more detailed breakdown of how the data is handled within the Amazon Web Service (AWS) and then imported into the VE follows. The chapters focus then moves onto the VE and specifically the Graphical User Interface (GUI) within it. This project led to the successful display of context-sensitive data within a VE, in a way the user found both intuitive and easy to interact with. The smart frame project was criticised for its lack of realism, which was taken into greater consideration for the VE of the DAAC.

The DAAC seeks to use MR as a tool to address some of the current issues within nuclear manufacture. To achieve this the DAAC creates a VE (Figure 1.1(b)) that is used as digital twin of the real manufacturing cell (Figure 1.1(a). Data is captured from within the real manufacturing cell and displayed within the VE. Using room-sized tracking technology, the movements of the manufacturing cell users are displayed by avatars within the VE. While projectors mounted on the frame project out work instructions from the VE back into the real manufacturing cell.

Importantly this type of simulation differs from current Nuclear Simulators such as L-3 MAPPS (L3, 2015). L-3 MAPPS can simulate reactor controls, power core creation, and many other engineering functions surrounding the construction of reactor systems. However it does not digitally twin a manufacturing cell, or track individuals movements within a manufacturing cell. In fact the two simulations have very different use cases.

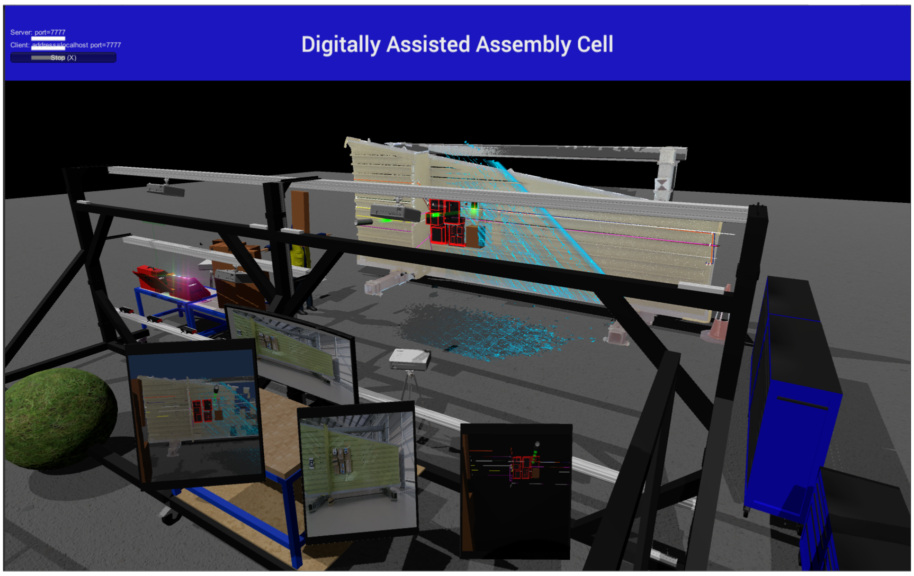
The users within the manufacturing cell are equipped with Android devices (Figure 1.1(c)) that allow them to see and interact with the VE and its users. This includes the ability to control the projected work instructions and a Voice over Internet Protocol (VoIP) system allowing them to talk directly to users within the VE.

The VE is hosted on a network that allows other users to connect and join the VE from anywhere in the world. Users can use Android devices, desktop computers or VR headsets as different platforms to join the VE. Each method has a different level of immersion, ranging from a one to one scale VR experience to a simple GUI tablet interface. All platforms support VoIP and allow for users to interact with each other and parts of the VE itself.

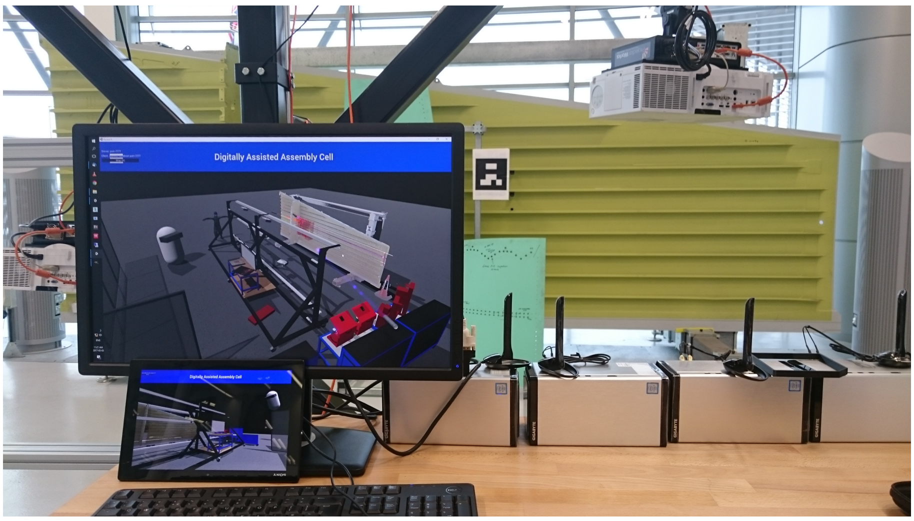
The concept of the DAAC is introduced in Chapter 7 with the stated aim to capture human activity, duplicate it within a VE and provide a feedback loop between the real and the virtual environments. To do this a bi-direction link is created between the virtual and real spaces. The chapter discussed the initial research into using Robot Operating System (ROS) and multiple Kinect sensors for tracking humans. An overview is then presented of the tracking space, virtual environment, feedback loop, human interaction and cell data sources. Diagrams of hardware requirements and networking implementations are presented to explain both how users were tracked within a manufacturing cell, and how data was projected into that same real manufacturing cell from its Digital Twin. The data sources, and their display within the VE are discussed to further expound on the idea of the VE representing a Digital Twin of its real counterpart. The chapter concludes with a final overview of the hardware configurations that make up the DAAC.

Chapter 8 is a more in-depth look at the separate systems that combine to form the DAAC. Starting with the backbone of the DAAC systems, which is the network communication layer. The chapter also goes into more detail surrounding the tracking software and methodologies provided by Ulm University. The Ulm Fusion tracking system is made up of several software abstraction layers which are detailed, before the skeletal data frames are discussed, to better explain the methods used for the import of the user tracking data. A brief section then details the use of a separate Kinect sensor to provide both RGB and depth sensor data for meshing within the VE. This leads into a discussion of the use of CAD data and laser scan point clouds and their respective import pipe lines, to make them viable for use within the Unity3D engine. The Networking and associated multiple user challenges are covered alongside the different user interfaces and GUI element required to support cross-platform access to the VE. The data sources section concludes by detailing the use of avatars as representations of users within the real manufacturing cell. The chapter then focuses down on the individual user, discussing the necessary code to provide user functionality in VR, desktop and Android. Such as VoIP and the ability to interact with other users and objects within the VE. Moving on the use of VoIP as a feedback mechanism, the use of

(a) The real world DAAC within the Advanced Manufacturing Research Centre (AMRC)



(b) The virtual DAAC showing work instructions, virtual tablets and real time sensor mesh



(c) The Android and desktop client mounted within the real world DAAC

Figure 1.1: A comparison of the virtual and real instances of the DAAC

projectors to re-project data back into the real cell is then discussed. The use of culling masks, created from objects grouped in layers, is explained as an effective mechanism for re-projecting select amounts of data back into the real manufacturing cell. The next sections concern the interaction of users with the system from each platform respectively. User interaction within the real manufacturing cell, and the use of the Android OS and tablets is discussed, alongside relevant features incorporated into the Android GUI. The VR interactions are detailed, with a focus on handling context-sensitive menus. Followed by an explanation of the use of events to handle multiple users interacting with a single objects within the VE. The underlying methodology of how the VE objects, cameras, projectors, tablets, laser pointers, and work instructions were created and implemented are discussed before the conclusion of the chapter.

In Chapter 9 the RE proposes that such a system will assist the nuclear manufacturing industry by allowing for manufacturing workers to:

- Train within a VE which is a digital twin of the manufacturing cell
- Train in a real manufacturing cell with remote assistance from experts in the VE
- Capture best practise of existing manufacturing experts for others to review
- Data capture to enhance the manufacturing *Quality Chain*
- Allow for higher worker utilisation through *Cross-Skill*
- Aid in rapid reconfiguration of the work environment and production routing
- Allow remote access to the manufacturing environment

The thesis concludes in Chapter 10 with a discussion on how the project could be used to address previously highlighted challenges within UK nuclear manufacturing. Finally, the RE reflects on what future enhancements could provide and further benefit, and which enhancements could increase the value of a system like the DAAC.

CHAPTER 2

Background

This chapter reviews the history of the nuclear industry within the UK, before identifying and discussing the current challenges within manufacturing for the nuclear industry. The chapter focuses on:

- The issues surrounding the supply of high value low volume components to the nuclear industry while ensuring that tight conditions of supply and regulatory controls are met.
- The requirement for highly trained staff to allow for rapid process change.
- The use of specialist equipment and a regime of zero defects.

The chapter concludes with a summary of the challenges associated within the nuclear industry.

2.1 The UK Nuclear Industry

Beginning with an outline of the history and subsequent resurgence of interest and opportunity within the UK's nuclear industry this section exposes the challenges that this resurgence has created; an expanding market, high levels of competition based upon price and quality, and a large skills gap are the main issues that need to be addressed if the UK nuclear manufacture industry is to succeed.

2.1.1 Generation 1 Reactors

The UK Nuclear Industry began in 1953 when the UK government announced it was starting a civil nuclear reactor program. This was followed shortly by the creation of the United Kingdom Atomic Energy Authority (UKAEA) in 1954, which oversaw the commissioning of the world's first civil nuclear power reactor at Calder Hall. This was of a Magnox reactor design, so called for its use of Magnesium Oxide in its fuel cladding. The design was used in a further ten reactors across the UK, eventually providing 10% of the UK's electrical supply. The Magnox reactors were an adaptation of previous reactors used to create plutonium (Appendix A.1). They used a graphite core moderator and carbon dioxide coolant. The reactors were also used to create fissile material for the UK's weapons program (UKAEA, 2014). Out of the 26 Magnox reactors built in the UK, none remain in service and all are now being decommissioned by Magnox Ltd (MAGNOX, 2014). The last reactor at Wylfa 1 was scheduled to shut-down in September 2014, and was finally shut down in December 2015.

2.1.2 Generation 2 Reactors

The UK would have to wait until the advent of the Advanced Gas Reactor (AGR) in 1983 for its first solely civil nuclear power plant. The AGRs, although similar in design to the Magnox reactors, operate at a much higher temperature, use enriched uranium fuel, and have a higher fuel burn-up so enjoy greater efficiencies. Despite being proposed in 1964 in a white paper titled 'The Second Nuclear Power Programme' (COLERAINE, 1964), unfortunately, in part due to these design advances, the first plant was not operational until 19 years later. The final AGR reactor design was commissioned in 1989, bringing the total units to 14 (Table 2.1). In 2014, life extensions were granted to the first reactors on the Heysham site, allowing them to operate until 2019 (EDF-ENERGY, 2014a). Following this the remaining AGRs have also been granted life extensions. This provides valuable opportunities for businesses and industry to capitalise on by either decommissioning the AGRs or working to validate them for further life extensions.

Table 2.1: Civil Nuclear Power Reactors operating in the UK (WORLD NUCLEAR ASSOCIATION (WNA), 2017)

Plant	Type	Present capacity (MWe net)	First power	Expected shutdown
Dungeness B 1&2	AGR	2 x 520	1983 & 1985	2028
Hartlepool 1&2	AGR	595, 585	1983 & 1984	2024
Heysham I 1&2	AGR	580, 575	1983 & 1984	2024
Heysham II 1&2	AGR	2 x 610	1988	2030
Hinkley Point B 1&2	AGR	475, 470	1976	2023
Hunterston B 1&2	AGR	475, 485	1976 & 1977	2023
Torness 1&2	AGR	590, 595	1988 & 1989	2030
Sizewell B	PWR	1198	1995	2035
Total: 15 units		8883 MWe		

The last civil nuclear reactor to be built in the UK was the Sizewell B Pressurised Water Reactor (PWR). Commissioned in 1995, the reactor had suffered from severe delays in both planning and building. A PWR uses enriched uranium oxide fuel, zirconium cladding, and a two loop water cooling system. Although further PWRs were planned for the UK, by May 1995 it was deemed that any new build would not receive public sector support (WNA, 2013). The delays were primarily over the spiralling costs associated with the creation of the plant, and whether a state-owned company should pay for them. This sparked a long running public inquiry which caused further costs and delay (GOVERNMENT, 2014). It stands as a stark lesson for the industry that it must deliver on time and on cost. 1995 marked the start of a significant hiatus in the UK's nuclear build program, that has only recently been broken when work started on the Hinkley Point C Evolutionary Power Reactor (EPR). (ENERGY DE FRANCE (EDF), 2017).

2.1.3 Generation 3 Reactors

Technological advances and subsequent design improvements have led to the development of generation 3+ reactors such as Hinkley Point. However the length of time between the previous generation of AGRs & Sizewell B, and the proposed new generation 3+ reactors has been sufficiently long enough to create a large skills gap. In the U.S.A. the Department of Energy assessed that in

2010 75% of its staff were eligible for retirement (WOGMAN et al., 2005). In the UK a similar situation exists, where a large gap exists between the older and younger generations of nuclear scientists and engineers. It can be of little doubt that these skills need to be transferred and then enhanced to modern day standards by the upcoming generation. Solutions will need to be found to train and educate the new generation.

2.1.4 Public Support for the Nuclear Industry

The nuclear industry has suffered a variety of setbacks over the years (IAEA, 2011) causing both loss of life and a loss of public approval. The most recent of these was the detonation of hydrogen in reactors at Fukushima in 2011, which led to the destruction of reactor units 1,2 and 3 (ONR, 2011). Despite this, in the UK the number of people in opposition to nuclear power dropped 7% from 2005 to 2013 (UKERC, 2014). The increased understanding and public awareness of climate change, energy security and fossil fuel use, has triggered a rethink of nuclear power globally (BICKERSTAFF et al., 2008), creating what is being called a ‘nuclear renaissance’ in both the UK and U.S.A. (GRIMES et al., 2010) . Increasing public awareness of the likely role that the burning of fossil fuels has, in climate change has led to a greater, if reluctant, acceptance of nuclear power as part of the UK’s future energy mix. If a nuclear renaissance is to take place within the UK, public perception is a key issue that needs to be addressed (GOODFELLOW et al., 2011). The use of modern technologies such as Mixed Reality (MR) in marketing (VAN KREVELEN et al., 2010), and in education (PSOTKA, 1995), could have a considerable affect on both public perception and understanding of the nuclear industry.

2.1.5 Reactor Design

In the UK, around a quarter of the existing power generating capacity is scheduled to be decommissioned in the next ten years. Replacing the existing power plants and their infrastructure has become a priority for the government and its agencies. Research into the post privatisation of the twelve UK energy firms (DOMAH et al., 2000), shows that cost has become the predominant factor in deciding the type of reactor that vendors would choose to build. With high fluctuations in fossil fuels over the last decade, it is argued that a liberalised wholesale market will not lead to a suitably efficient and diverse energy mix (ROQUES et al., 2006). The government white paper on energy states that

> “the current market arrangements will not deliver investment at the scale and pace that we need.” (DECC, 2011b)

One of the major hurdles for nuclear energy is that it requires a large long term investment in both the power plant itself, and the specialised infrastructure that supports it. To achieve this the Government has secured further cooperation and funding with China, which is aimed at using nuclear power plants to

> “reduce emissions and enhance energy security” (DECC, 2014a)

The cooperation and funding agreement has led to China becoming involved with the funding of the EDF EPRs at Hinkley Point C. The equity will be split between EDF, China General Nuclear (CGN), China National Nuclear Corporation (CNNC), and AREVA. The way was also paved for Chinese companies to both own and operate a Chinese-designed nuclear power plant in the UK.

This could lead to a faster rate of expansion of the UK's nuclear new-build program, and has already led to a Memorandum of Understanding (MoU) being signed on the 18th of June 2014 between Rolls-Royce and CGN (Rolls-Royce, 2014). The MoU will see Rolls-Royce working more closely with CGN in the area of civil nuclear power, both domestically and internationally. This is in addition to the fact that

> "Today, Rolls-Royce supplies safety-critical instrumentation and control technology to more than 70% of nuclear reactors in operation or under construction in China and Emergency Diesel Generators to almost 40 per cent." (Rolls-Royce, 2014)

This is a clear signal from Rolls-Royce of their intent to expand within the nuclear sector, and could present further opportunities for exporting successful solutions created for the UK market abroad. To achieve this all parties are interested in a reduction in cost, and on delivering on time, so as to avoid a repeat of the issues that surrounded Sizewell B. One of the major issues that is associated with the Sizewell B Nuclear Power Plant (NPP) is that of standardisation. As a single plant, most components of the plant were built as a one off, whereas if a fleet of reactors were to be built, then standardisations of parts and components could be created. Rolls-Royce is hopeful that the go-ahead of their Small Modular Reactor (SMR) platform, could provide an opportunity to create a new set of standard components for a fleet of their SMRs.

In July 2011 the UK government released its National Policy Statement (NPS) for Nuclear Power Generation EN-6 (DECC, 2011a), within which suitable sites for the placement of the next generation of reactors were proposed (Figure 2.1). These included areas Bradwell, Hartlepool, Heysham, Hinkley Point, Oldbury, Sizewell, Sellafield, and Wylfa. As figure 2.1 shows, all of these are current nuclear licensed sites. In total, four sites have been bought by three different groups; EDF Energy have bought sites at Hinkley Point and Sizewell; Horizon Nuclear Power (owned by Hitachi) have bought sites at Wylfa and Oldbury, and Nugen (owned by Toshiba (Westinghouse)) have bought the Moorside site at Sellafield.

The reactor designs being considered are the Advanced Boiling Water Reactor (ABWR) designed by Hitachi-GE Nuclear Energy Ltd (Hitachi-GE, 2014), the AP1000 PWR designed by Westinghouse (Westinghouse, 2014), the EPR designed by EDF and the HPR1000 designed by CGN. The ABWR is still undergoing Generic Design Assessments (GDA) set by the Office of Nuclear Regulation (ONR) and Environmental Protection Agency (EPA) for use in the UK (HSE, 2013). The EPR and AP1000 have passed the GDAs required. AREVA has continued through the process and been granted a certificate of 'Design Acceptance Confirmation and Statement of Design Acceptability'. With the initial licensing milestone passed, the first reactor is set to be built on the Sizewell site, with completion expected around 2023. In March 2017 NuGen also received acceptance of the AP1000 design for their Moorside site (ONR, 2017a), and hope to be full Nuclear Site Licensees for their ABWR site in Wylfa by the end of 2018 (ONR, 2017b). General Nuclear System Limited (GNS), a company created through the collaboration of EDF and CGN, have submitted their designs for the UK HPR1000 to the ONR GDA process. It is expected to take five years to process, and the final plants are scheduled for the Bradwell site in Essex (GNS, 2017). The designs have entered the third of four stages of the GDA process, but there is still no news on when construction or operation of the any reactors built on the Bradwell site could be. The reference plant in Fangchenggang is still under construction, and it is foreseen that any reactor built based on this design will involve years of investigative works and public consultations before detailed planning applications could be made (CGN, 2018).

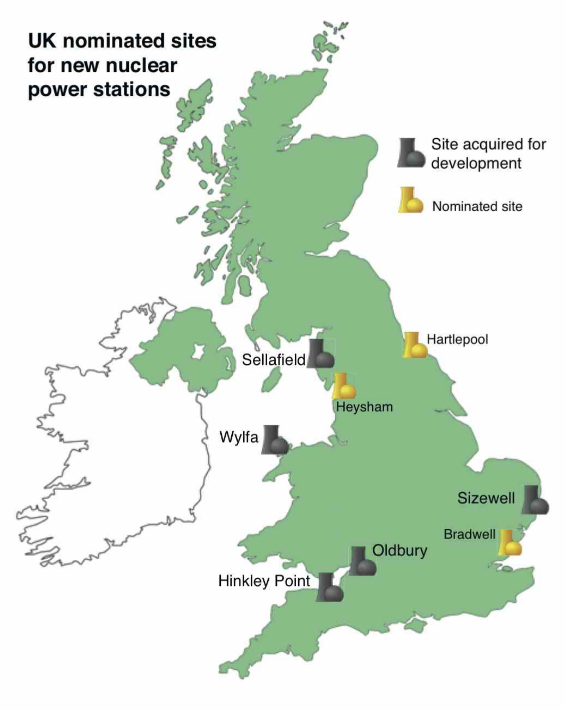


Figure 2.1: Current and planned civil nuclear sites in the UK (DECC, 2014b)

A possible 16GWe could be created by the new-build program from a possible total of 13 new reactors. EDF-Energy is proposing to build two AREVA EPRs at Hinkley Point, Somerset, and two at Sizewell, Suffolk (EDF-ENERGY, 2014b). Horizon Nuclear Power, owned by Hitachi, is planning to build two or three 1.3GWe ABWRs at both Wylfa, Anglesey and Oldbury, Gloucestershire (HORIZON, 2014). NuGen is intending to build up to 3.6GWe new capacity at the Moorside site at Sellafield, Cumbria (NUGEN, 2014). With all three vendors stating intent to source production and services from the UK market the UK stands to gain considerably in terms of both GDP and employment (NAMRC, 2014). The agreements for manufacture within the UK in place with existing companies do not include major island components. This is in part due to the limited manufacturing capabilities within the UK, but also because the majority of Internet Protocol address (IP) is locked up in the primary circuit components. A Nuclear Industry Association (NIA) report suggests that upwards of 30,000 jobs could be created by the new build program (NIA, 2011). The large number of sites will create a boom for the nuclear industry and jobs across the country. However, this requires the training and education of staff. There is also a high variance in the reactor designs, which will mean that trained staff will be limited in their ability to cross easily between reactor designs and companies.

2.1.6 Small Modular Reactors

Post generation 3+ reactors, the future of the nuclear sector may lie in the design and development of SMRs. These reactor designs have been proposed for use around the UK and for export to foreign countries. In 2016 the UK government launched a planned £250 million pound investment in research for SMRs, which included a design competition to help identify the best design for the UK (Department for Business, Energy & Industrial Strategy (DBEIS), 2016).

Rolls-Royce are competing with Chinese state firm CNNC, US firm NuScale and the US-Japanese partnership GE-Hitachi. Toshiba's US nuclear arm Westinghouse was also among the bidders but is now unlikely to proceed after filing for bankruptcy in March 2017.

Table 2.2: Reactor designs submitted for UK viability competition (NAMRC, 2017)

Company	Type	Projected Capacity	Features
CNNC	PWR	100MWe & 310MW thermal	The ACP100 is an adaption of the Westinghouse AP1000 design
GF Nuclear	integrated PWR	100MWe	Developed with Korea Atomic Energy Research Institute (KAERI) the SMART reactor is already licensed in Source Korea
Moltex Energy	Stable Salt Reactor	150MWe	Designed for modular deployment
NuScale Power	PWR	50MWe	Designed for modular deployment of clusters up to 12 per site
Rolls-Royce	PWR	220MWe scaling to 440MWe	The modular reactor design allows transport by truck, train or barge.
Tokamak Energy	Fusion	N/A	The ST40 design is stated to become commercially viable by 2030
Urenco	Pebble Bed	2 x 520	Working with Amec and Atkins on an ultra-small design called U-Battery
Westinghouse	PWR	225MWe	Based on its AP1000 design

A number of the reactors above are considered to be Advanced Modular Reactor (AMR) reactors, these are reactors that do not rely on tried and tested technologies taken from existing PWRs. Instead they rely on data from existing scaled down test reactors. This means that their time to market is unknown, and in an entirely different ball park to that of SMRs. Technologies involved in Molten Salt Reactors (MSRs), Pebble Bed and Fusion reactors are outside the scope of this thesis. They are fascinating in their own right, and are no doubt the topic of many other Doctorate of Philosophys (PhDs) and Engineering Doctorates (EngDs). The scope of this thesis will stay with

SMR, PWR, and nuclear fission reactors, and the manufacture of their components.

The 2014 NNL report predicts a potential global SMR market of 65–85GWe by 2035, valued at £250–400 billion; and a UK market of around 7GWe (NATIONAL NUCLEAR LABORATORY (NNL), 2017).

The strike price (agreed price per MWh) is estimated to be much lower when larger counterparts estimated at £60/MWh for the Rolls-Royce SMR, in comparison to the £92.50/MWh agreed for the EPRs at Hinkley Point C. The construction costs are also thought to be much lower due to modular building techniques, allowing for large parts of the construction to take place off-site in factories.

However, it is this method that may have exacerbated the issues that Westinghouse have had building their AP1000 reactors in the USA (REUTERS, 2017), a difficulty which may be insurmountable as Westinghouse filed for bankruptcy in the USA on 29 March 2017. Its parent company Toshiba Corp has also been in trouble recently, although the sale of its chip industry and a recent issue of shares may help improve its situation (FINANCIAL TIMES (FT), 2017).

Despite this the American Department of Energy (DOE) sees SMRs as a part of their future energy generation mix;

> ‘Advanced Small Modular Reactors (SMRs) are a key part of the Department’s goal to develop safe, clean, and affordable nuclear power options” (DOE, 2017)

Although in the UK the situation is less clear cut as;

> ‘SMRs will need to deliver energy cost-competitively if they are to play a part in the UK’s future energy mix. As well as securing low-carbon energy, government is also committed to keeping down the cost of that energy for consumers, so there is a key challenge there for the nuclear industry as a whole and for SMRs” (WINTLE, 2017)

At the time of research, geo-political issues between the UK and France played a large role in the predicted future of power generation from nuclear fission plants. The nature of this and its consequences are outside the scope of the current research. These issues have been compounded by the government’s lack of communication over the results of their SMR design communication, which has left vendors in a difficult position as to whether to move forward with their designs and into the licensing stages.

Although SMRs seem to be a part of the future of nuclear power generation, exactly what part this is, is as-yet unclear, especially in the UK.

2.1.7 Conclusion

As can be seen, there is a long history of innovation within the UK nuclear industry, which at one point led to the UK having one of the greatest variety of nuclear reactors in service. The current fleet of AGR reactors are reaching End of Life (EoL) in the next decade. and the government are investing in a new generation of civil nuclear power plants as replacements. The introduction of the previous two generations of nuclear power plants was fraught with delays. With public scrutiny of the nuclear industry increasing due to recent severe accidents, it is important that the installation of the next generation of power plants is as accident and delay-free as possible. This will require a combination of effort from across the nuclear industry, within which the role of the nuclear manufacturers will be to provide nuclear grade components on time and to specification. To meet these specifications new testing methods, and higher fidelity simulations of this data will need to be performed.

There has been a significant gap between the installation of the last civil nuclear reactor in the UK and the proposed installation of the next generation. This has created a long gap between manufacture periods for the nuclear industry, meaning UK manufacturers will have to adapt to meet an increase in demand. This will include the training of a labour force that has never worked under nuclear manufacturing conditions nor adhered to its environment of stringent regulation. Although in today's global market, it is expected that many components will be manufactured abroad, it is also likely that new modern manufacturing facilities will need to be created in the UK to cope with a wide range of high quality components. A single plant may be subject to multiple manufacturing codes, nuclear pressure vessels must conform to ASME III codes, whereas conventional pressure vessels (vessels containing no radioactive materials), heat exchangers and pressure pipes fall under the ASME VIII code. This will require high quality manufacturing techniques, training and associated technology to be deployed across the supply chain. Companies in the UK will need to compete on quality and price within a global market, which has an established presence and existing expertise (DCNS, 2014; Skoda, 2014).

This section has shown that in the next decade the nuclear industry in the UK could capitalise and develop in four major areas:

- Decommissioning the oldest EDF-Energys' AGRs
- Providing technical assistance and safety verification for the life extension of the rest of EDF-Energy AGRs
- Manufacture for the new nuclear build Gen-3+ reactors
- Design and build of SMRs

The section has also identified the challenges that the nuclear fission industry must overcome to benefit from such development.

- The skills gap
- The variation in plant design
- The limited number of reactors being built per design
- Public support for the nuclear industry as a whole

A discussion of the challenges relating more specifically to the nuclear manufacturing industry follows.

2.2 Nuclear manufacturing

Manufacturing for the nuclear sector imposes a series of challenges on companies. This section will identify areas that contribute to high manufacturing costs and the importance of zero defects.

High value, low volume: As the current generation of nuclear power stations generate Giga Watts of electricity, the total number of plants required to meet the base load demands of the UK energy sector is relatively low. With a proposed maximum of 13 new reactors being built in the UK, the number of full plant sets for the UK market will be limited. This creates an industry where the demand for parts can be considered low volume. To be cost-competitive the low volume of the components requires a flexible production facility, which can accommodate rapid process changes, changes of jigs, and resetting of parts. It will also require that the staff have a high level of versatility, or that the staff are contracted only for the duration that their specialisation is required. The staff will also require refresher training for tasks which occur significantly less frequently than others (GLOBERSON et al., 1989; SCHULTZ et al., 2003). For parts which are particularly large and/or complex and have long lead-times this would be even more important. All of which further increases the cost of manufacture(LIVESEY, 2006).

Discussed later is the fact that the quality required of the components within a nuclear facility meet stringent testing standards and have a traceability from ore. These high standards require a great deal of Quality Assurance (QA), which in turn can lead to a what can seem like an exponential growth in paperwork and costs. These high standards are in part due to the consequences of failure in mission critical components in a reactor being potentially catastrophic. Therefore the associated value of these components must be considered high. That is to say the amount of investment to ensure they are as safe as possible, or that the risk of failure is As Low As Reasonably Possible (ALARP) is higher than it would be for many other industries. The consequences for a facility that is manufacturing nuclear components are serious and costly. Firstly, the high value components require proportionally high levels of conformance. Meeting this, demands a high technical ability in staff and equipment, both of which increases the cost of manufacture. Re-work or non-conformance is also more costly, so a policy of zero defects is often implemented and great care exercised to minimise damage to components.

Component range: The range of components required for a nuclear power plant sets it apart from many other manufacturing industries. The size of products can be, in the case of a steam generator, 20m long with a diameter of 4m and a weight of 580 tonnes. Contrasted with a former baffle plate bolt, which is just over 8cm long (P. M. SCOTT et al., 2000) shows a large range in product size. A steam generator can contain up to 16,000 tubes, figure 2.2, showing a high component count and high complexity of assembly. The steam generator requires large and heavy forgings, which require correspondingly large and therefore expensive machines. In contrast the manual handling tasks required in loading the tubes into the steam generator through thousands of baffle plate holes requires less capital expenditure, but is still high cost due to the level of expertise required. The transition between heavy machine work and delicate manual labour, requires complex movements of the parts and apparatus between safe working areas.

Figure 2.2: Steam generator lifting & tube loading (F-M, 2014)

Handling: The low volume of manufacture means that any complex or specialised transportation tasks are not carried out regularly. The low frequency of task will mean staff are not well practised in carrying out the task, thus leading to a greater risk of failure (GLOBERSON et al., 1989; SCHULTZ et al., 2003). This further increases costs through insurance premiums and the requirements of a large safety case. The size of some of the components will also increase expenditure on cranes and other handling apparatus, as well as require special ingress and egress methods, such as large doors and transporters. The size and weight of the components also affects the time taken to move them around the shop floor. The higher the frequency of moves the component must undergo the greater effect this has on the overall lead-time of the component and the greater potential for damage.

Conditions of supply: The specifications for nuclear manufacture often contain very stringent conditions of supply. One of these is cleanliness; to prevent materials damage and foreign objects entering any of the coolant loops of the reactor, which could lead to damage to critical components and could affect affect the fine balance of chemistry within the coolant systems and the Instrument & Control Systems (ICS). The stringent cleanliness conditions require that cross-contamination does not occur at any point in the manufacturing process. This can require that machines and cells are cleaned down between processes on different components, which in turn adds to the production time. In some cases it may require that separate machines are used for components made of different materials, and even that components are assembled in clean rooms. On the handling side it means that strict control measures need to be in place to control where and how materials and components are used and moved.

The manufacturing processes themselves are strictly controlled. The American Society of Mechanical Engineers (ASME) produce sets of codes which contain the methods and processes of manufacture that must be adhered to (ASME, 2013). The codes reduce the methods of manufacture available to a producer, and restrict how those methods are carried out. This reduces the possibility of innovation within the nuclear manufacturing sector, and the possibilities of using new and improved techniques, at least until they have been proven by rigorous scientific testing, and accepted by the independent bodies such as ASME. However that is not to say that there is no innovation, novel ideas are still allowed so long as they fall within defined codes and processes.

Only processes that influence the properties, product or changes for approved traceable routes require a resubmission. These new process are the ones that will require a "code case submission" or standards improvement.

Fine tolerances: ASME and other regulations require very fine tolerances on both the welding and machining of the components. This not only ensures that component safety margins are maintained within the plant, but also allow for finer levels of precision engineering to be used within the design phase. This can be useful, when designing to manufacture, as the processes are already clearly defined and the tolerances set. However the reverse is true when manufacturing the components, as fine tolerances require more exacting machines, and defined processes reduce the flexibility within a process. The codes also specify the method of Non-Destructive Testing (NDT) to be used, required to make sure that the products conform. The examination standard for Non-Destructive Evaluation (NDE) and the supplemental customer standards are what define the level of acceptance.

High specification materials: Within nuclear manufacturing a wide range of high specification metals are used. The high specification of the metals poses a challenge for cross contamination, but also requires unique research for manufacture as the use case and manufacture processes required can be unique to the sector. This is particularly true with zirconium alloys used in fuel rod manufacture, as it is highly flammable once in powder form, and also exothermic enough that water is unsafe to use as an extinguishing method as it will lead to hydrogen production. Another issue similar to cross contamination, is of joining different metals together to form a single component. This requires specialised joining techniques and skilled welding. This will allow the vessel to operate at the required range of temperatures without the join breaking apart. The melting points of the two metals needs to also be managed correctly as, if heated simultaneously, one metal could melt significantly quicker than the other. It is also important that the metals are stored and combined in a way that means that galvanic corrosion does not take place.

Documentation and traceability: The adherence to stringent codes and the requirement of high tolerances on the products, means that the ability to trace the progress of the product and record data about the processes that took place is important. This generates a large amount of documentation that can take up valuable hours of production time and can result in delays. One of the positives of the documentation is traceability and increased accountability for the product's life cycle, requiring an effective paper trail of what has happened to the product, describing when and how it is created. This allows a kind of forensic analysis, when non-conformance is discovered in a product or material. Finding the exact process or point where the non-conformance was caused, allows actions to be taken where they are needed. The records themselves will contain information such as process cycle times, who worked on the process, when it was completed, NDT results, etc.

Lead-times: All of the challenges above have the affect of increasing the production lead-time of any product. This can be further increased by customer demands for hold points. Hold points are where production on a product is stopped for a period of time so that external parties can inspect and verify its conformance to specifications. The outside parties could be the customers themselves, or members of a professional body. The hold points are not subject to a company's own control, as its completion is dependant on other companies' production schedules.

2.2.1 How is manufacturing in the nuclear industry unique?

Manufacturing for the oil & gas sector has many similarities with manufacturing for the nuclear sector. This is due to the fact that both industries manufacture product sets for power generation facilities and often deal with large and unique products. The predominant difference between manufacturing in each sector is one of quality and performance requirements (SHARP, 2014). When manufacturing components for nuclear power plants, the quality and performance requirements are much higher, as the potential consequences of component failure could be more catastrophic. Although a component failure in the oil & gas sector may lead to an accident as damaging as one in the nuclear sector, the long term consequences of an accident in the nuclear sector can be far greater (SMITHSONIAN, 2014). The lasting effects of nuclear incidents involving the release of radioactive materials are well documented (FUGAZZOLA et al., 1995; NAKANISHI et al., 1991; SALBU et al., 1994; YASUNARI et al., 2011). The damage to society and the cost of recovery are far greater post nuclear accident, and to compensate for this, the requirements for quality are much higher. It is the consequences of failure that drives greater levels of control over the processes, the consumables and the culture in a nuclear manufacturing facility (INPO, 2014).

For example, the control over materials is also very specific in oil and gas applications, where commercially available grades of steel must fit the specifications for pressure vessels and piping. When manufacturing in the nuclear sector, the steel must be carefully tested to ensure that the composition, specifically the impurities, is within the levels as defined in the specification (ASME, 2013). This is because impurities within a material can lead to Environmental Assisted Cracking (Environmentally Assisted Cracking (EAC)) (ANDRESEN et al., 1988). During normal operation, when impurities such as chlorides, fluorides and sulphates, are relatively mobile with the metal. Impurities tend to segregate to grain boundaries and interstitial sites, causing embrittlement, which can lead to the early onset of cracking.

Ensuring that materials conform to an appropriate standard is important to ensure that mechanical performance criteria can be met. Additionally, the presence of certain impurities can contribute to alternative failure mechanisms, such as enhanced corrosion, through-life irradiation embrittlement, and environmentally assisted cracking (such as stress corrosion cracking or environmentally assisted fatigue). Impurities such as copper and phosphorus are known to lead to embrittlement problems in low alloy steels (such as those used in reactor pressure vessels), and are examples of elements on which contents are tightly controlled.

Another area that requires tight controls to ensure materials performance is manufacturing. For example, when certain stainless steels are welded, a process known as sensitisation can occur, in which chromium carbides of the constituency $Cr_{23}C_6$ form at the grain boundaries, reducing the amount of free chromium available in the microstructure to form a passive layer (an adherent chromium oxide layer) that forms on stainless steels and is the reason for their excellent corrosion resistance. This leads to an enhanced susceptibility to intergranular corrosion and EAC. Sensitisation can be controlled through either an 0.03wt%C carbon content or the inclusion of additional elements in the metal, such as niobium which preferentially forms carbides.

Unique to the nuclear industry is the restriction on Cobalt. The naturally occurring isotope Co^{59} has a high cross section of absorption for neutrons at the thermal energy ranges common in reactors (Figure 2.3), and can therefore activate to become Co^{60} which is radioactive. With a half-life of over five years, and a potentially very harmful gamma emission of 7.478 KeV, Co^{60} can increase the time period before components can be reprocessed, replaced, or handled (IAEA, 2014). Furthermore once irradiated it can pose serious harm to maintenance workers. To minimise the presence of impurities in the materials, both the supply and testing of materials are tightly controlled.

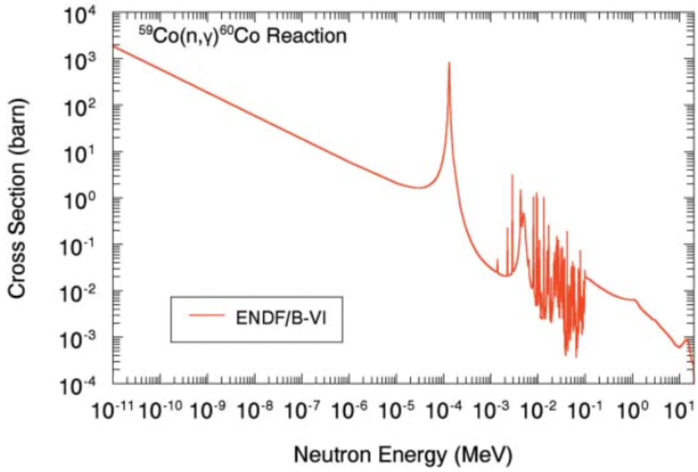


Figure 2.3: Energy dependence of the cross sections for the $Co^{59}(n,\gamma)Co^{60}$ reaction (IAEA, 2009)

The consumables used in a facility also need to be closely controlled: a list is created of consumables such as Personal Protective Equipment (PPE), that are safe to use within the facility. This list is also job specific, as a consumable used on high carbon steel products, may not be suitable for a stainless one. This separation of consumables incurs greater cost, but also helps reduce cross contamination. Furthermore the controls put in place to maintain high levels of cleanliness are expensive and can add significant time to production.

The controls also extend to the culture within the facility, as it is imperative that the staff understand the requirements made of them. This requires that staff are aware, not only of how to maintain segregation and control contaminants, but also why they are doing it (T. Lee et al., 2000). The additional training requirements imposed on all employees to maintain correct work ethics and cleanliness further add to the cost of manufacture.

These controls are created to ensure that component failure, especially that of critical components, should not occur within the lifetime of a component (Onizawa et al., 2010). The controls implemented from the safety case analysis for a pressure vessel ensure that the ONRs stated failure rate of Incredibility of failure (IoF) of $1x10^-7$ per vessel year can be met.

2.2.2 Discussion

It has been shown that there is a high cost associated with nuclear manufacturing. In addition to the challenges identified in Section 2.1, the high cost of manufacture arises from the need for:

- High conformance to codes such as ASME
- Highly trained staff
- Rapid process change
- Flexible facilities and staff
- Diverse range of components
- Specialist equipment for handling, & manufacture
- Slow uptake of innovative techniques
- Documentation
- Long lead-times

The expansion of the nuclear industry would have driven growth in the manufacturing industry as it expands to produce the required components (UK-GOV, 2013a; UK-GOV, 2013c; UK-GOV, 2013d). The expansion was set to benefit large companies such as Rolls-Royce as well as smaller companies that would make up the supply chains (UK-GOV, 2013b).The high level of difficulty and large costs involved, create an environment where costs savings, and improvements to process, are desirable. The stringent regulations mean that clear guidelines are in place for both manufacturing processes and results. To ensure that the guideline conditions are met, process methodology and timings are recorded. This has the added benefit that it allows for processes like benchmarking and continuous improvement to be implemented.

2.2.3 The DAAC Cell

It is hope that the Digitally Assisted Assembly Cell (DAAC) can be used as a tool to help address all of these issues.

High Value, Low Volume The recording of human operations with a manufacturing cell can be of great assistance within a quality chain, and could be used to help reduce elements of human error. Futhermore the use of a Virtual Environment (VE) which is a digital twin of the real manufacturing cell allows of users to design and set-up new operations before they take place, this can then be extended to train in a facsimile of the manufacturing cell. This is particularly effective in in the bracket of low volume high value, as it is worth spending more time designing and setting up processes of manufacture to reduce the likelihood of future issues. With the low volume of manufacture there is little change to get it right on the next run.

Component Range The same benefits will allow for teams to design and train staff on a wide range of components before manufacture begins, and then use the DAAC system to assist the staff in real time while performing the manufacture process. This is especially useful when staff are changing component and manufacturing processes.

Handling The DAAC provides the ability to design and test handling techniques within a digital twin of the manufacturing environment, and then record and capture the movements and techniques performed in the real manufacturing cell. The handling methods can be designed and reviewed in the virtual, and then tested against the real before further design and process improvement processes can be carried out. An iterative process of improvement can be implemented where any expert can view either the recorded and planned process, or watch the real manufacture process in action from anywhere in the world.

Conditions of Supply The condition of supply might improve with the ability to trace any faults or errors back to the individual and process at the root of them. This root cause analysis is made easier by the capture of the manufacturing staff movements and actions while performing the process. The DAAC also makes it possible for anyone in any location to dial in and assist the manufacturers remotely at any point in the manufacture process. This could be particularly useful when performing critical tasks.

Fine Tolerances The DAAC does not record the tools or the tolerances the tools work to. However it does track the employees movements. This could be of great use when checking that full NDE processes have been completed correctly. Did the member of staff have access to all the areas that required NDE or does the manufacturing process need changing to allow better access? This could be discovered using the DAAC at any point, and then the solutions could be reworked within the VE, before being tested and observed within the real manufacturing cell.

Documentation and Traceability The human tracking could be added to root causes analysis methods and work within the quality chain.

Lead-Times These could be reduced by studying the movements and work flow of employees within a manufacturing cell. Best practise and optimised cell movements and work flows could be captured in one DAAC, reviewed and refined and then tested in any similar DAAC in the world. The optimisation is of human manufacturing processes has long been out of reach for all but the largest manufacturing companies, the DAAC provides a cheaper, smaller and more focussed platform for this.

High Conformance to Codes such as ASME The ASME codes and methods of production could easily be added to the work instructions within the DAAC, and then they could be delivered to the employee at the right time, in the right place for them to conform to them. On top of these the real time RGB camera allows for remote workers to check back and ensure that the codes were met. Although this holds little sway with ASME, it could prove to be useful internally within a company.

Highly Trained Staff The DAAC allows staff to be trained either within a VE containing the digital twin of the manufacturing cell, or within the real manufacturing cell. Staff can be trained while own their own, or assisted within either the VE or by users within the VE while they themselves are in the real manufacturing cell. The data collected from other staff working within similar cells can be used to assist in training of new staff or re-training of existing staff.

Rapid Process Change with the ability to connect to the DAAC from anywhere in the world then access to new process and changes in manufacturing techniques could be almost instantaneous. With remote assistance, staff can be retrained by trainers anywhere in the world.

Flexible facilities and staff The DAAC could be used to help cross-skill staff by projecting detailed work instructions onto the work pieces as and when they are required. Furthermore it allows for staff to dial in and assist whenever an employee might seek assistance when working on a new or complex operation.

The DAAC offers possible solutions to issues that face the nuclear industry, however it should be viewed as only part of a much wider set of solutions offered by Virtual Reality (VR). In Section 4.2 academic MR solutions to some of the above issues are discussed. Investigation into MR within this thesis will be focused on addressing these challenges. To do this it is important that an understanding of the challenges and limitations within MR are also understood. The next section details key definitions that are required for discussions on the MR technologies and applications.

CHAPTER 3

Definitions & Key Terms

This section provides definitions for some of the key concepts and areas of knowledge discussed within the thesis, expanding on well researched, and widely implemented areas of Virtual Reality (VR) and Augmented Reality (AR). Section 3.2.1, describes how 3D images are re-created for use in Virtual Environment (VE), followed by Section 3.2.2 discussing Immersion, its key concepts of Place Illusion (PI) and Plausibility Illusion (PsI). The paradigm of Data, Information, Knowledge and Wisdom is then discussed in Section 3.3, which leads into the methods data capture within manufacturing. Challenges surrounding the use of VEs as a means of displaying these data sets are discussed alongside early pioneering uses of VR. The chapter concludes with the use of simulations as a means of converting data into information with specific focus upon the use of Discrete Event, Continuous, and Monte Carlo Simulation.

3.1 Mixed Reality

Mixed Reality (MR) sits between two absolutes, that of an environment that is entirely real, and one that is entirely virtual (Figure 3.1(a)).

Between the two absolutes, seven categories of reality have been defined (Figure 3.1(b));

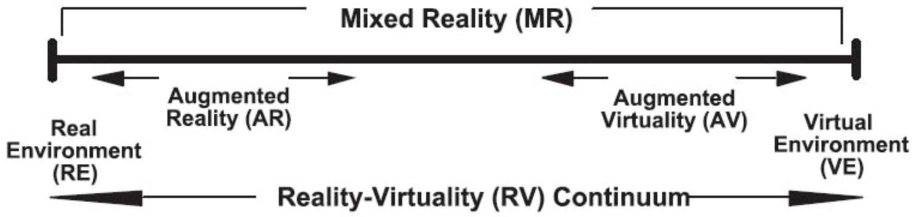
Real Reality - the entire Real Environment,

Amplified Reality - where objects use embedded computational resources to enhance their perceivable properties,

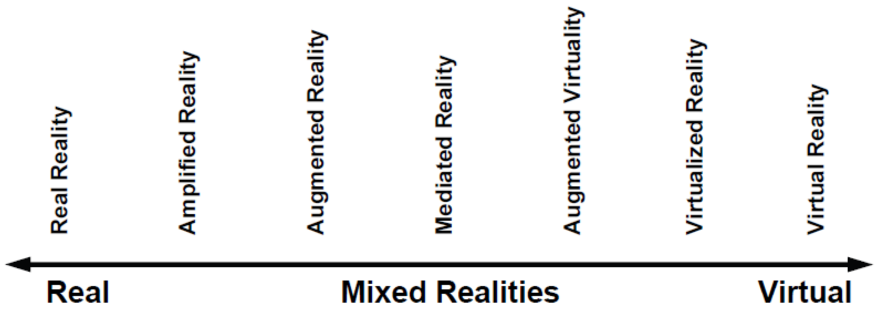
Augmented Reality - in which elements or objects of the real environment are augmented with the virtual elements,

Mediated Reality - where areas of the real world are changed by either being diminished or altered by partial virtual environments,

Augmented Virtuality - in which a virtual environment is augmented with the aspects or objects of the real,



(a) The 'Reality-Continuum' as proposed by MILGRAM and KISHINO (1994)



(b) Reality concepts ranging from Reality (left) to Virtual Reality (right) (SCHNABEL et al., 2007)

Figure 3.1: Comparison of different authors spectrum of Mixed Reality (MR)

Virtualised Reality - where the real environment has been captured and the computational model created to use within a full Virtual Environment,

and

Virtual Reality - in which the entire environment and its constituents are virtual.
Amplified Reality and AR have seen little use and suffered from a slow pace of development in comparison with their counterparts so will not be further defined (HALLNÄS et al., 2001). Virtualised Reality can be encapsulated into VR as they both operate in a fully virtual domain. In a similar fashion Mediated Reality can be discussed within AR.

3.1.1 Virtual Reality

VR was first coined by Jaron Lanier who pioneered its research in the late 1980s and early 1990s as head of VPL Research (CONN et al., 1989). Today VR is used in a number of different ways which can lead to confusion.

> "A computer-generated simulation of a lifelike environment that can be interacted within a seemingly real or physical way by a person, ***esp***. by means of responsive hardware such as a visor with screen or gloves with sensors; such environments or the associated technology as a medium of activity or field of study; cyberspace. " (OED, 2014)

The above definition is an accurate description of VR, of note, however, is the proviso that the environment can be interacted within a seemingly real or physical way. This is essential to something called the PsI and will be discussed later in the chapter. Viewed in stereo, a desktop monitor, TV screen, even IMAX will not create the Place Illusion required for immersion. All these display methods, especially when linked to responsive hardware, such as; a Microsoft Kinect, Leap Motion, Wii remote, all fit within the definition of VR. This allows environments such as Second Life, World of Warcraft and other Massively Multi-player Online games (MMOs) to fall under the genus of VR. The encompassing nature of the definition allows examples that do not support immersion to fall within it. A more concise definition of VR is required to incorporate solely the systems that create immersion.

> "Immersive Virtual Reality (IVR) is the science and technology required for a user to feel present, via perceptive, cognitive and functional immersion and interaction, in a generated environment." (VAN DAM et al., 2000)

IVR can be seen as a subset of VR, with the benefits of both PI and PsI.

> "A medium composed of interactive computer simulations that sense the participant's position and actions, providing synthetic feedback to one or more senses, giving the feeling of being immersed or being present in the simulation" (CRAIG et al., 2009)

3.1.2 Augmented Reality

> "the use of technology which allows the perception of the physical world to be enhanced or modified by computer-generated stimuli perceived with the aid of special equipment; reality as perceived in this way." (OED, 2014)

To expand on the previous definition, AR is when information from a VE is overlaid onto actual reality. This can be done by the use of a Head Mounted Display (HMD), or any screen that is capable of displaying computer generated imagery. Both methods are prevalent within the military, primarily within the air force, where they are used within a pilot's helmet, or projected onto the cockpit canopy. In this way, AR is the blend of virtual data with real world data which creates new knowledge for the user. For example, the locations of friendly aircraft hidden behind a cloud. The real world data can be part of a live stream, like the feed from a camera, whereby the virtual data is overlaid onto the live stream and viewed through another device. The measure of how much data from the real world is a direct overlay and how much is from a live stream is called directness (MILGRAM, TAKEMURA et al., 1995). The greater the level of data that comes from the overlay of the real world without intermediate streaming, the greater the directness can be said to be.

To align the virtual and real world data, so that they match up in a meaningful way for the end-user, a level of information about the real world needs to be present in the virtual. This is done by tracking aspects of the real world, either by direct measurement, such as location or positional tracking, or by image recognition. Real world knowledge and display is fundamental to the definition of AR. Without the interaction of the real and virtual worlds, the augmentation of ink letters on paper could be considered AR. The device or systems knowledge of the real world allows the virtual data to be added to the real world data, in a fashion that increases the user's understanding or knowledge of the real world.

3.2 Key terms

This section briefly outlines the key terms used in the thesis to provide sufficient background information. All the defined terms here, and later expanded upon in the Literature review (Section 4) are all interconnected. Without the use of Stereoscopy, the illusion of presence PI and plausibility PsI would be almost impossible. Without these key contributors to immersion, then the effectiveness of using a VE as a method of transferring information across to users to turn into knowledge would be diminished. Finally, without simulation, the data sources could not be processed, assembled and delivered into information for use within the VE itself (Sections 3.3 & 3.4). In this way MR is like a focal point for different data streams and information sources. The technology alone lacks discernible benefit until it is applied, and only the input streams combine to create discernible benefit to the user. Like all good tools, the more the user knows about it, the more they understand how it can be used to its fullest potential.

3.2.1 Stereoscopy

> " The art or practice of using the stereoscope" (OED, 2014)

Stereoscopy, derived from the greek word 'stereo' (that is relating to space), relates back to the first uses of the word in the English language, to denote the Stereoscope; a device that used two images to create the illusion of depth for the viewer. This method was first devised by Charles Wheatstone in 1832 (WHEATSTONE, 1838). The device used a mirrored system to align two images of the same scene to each eye. The images were taken with a slight spatial separation, relating to the inter-axial distance between ones eyes. When the two images were recombined they created the illusion of depth within the image.

A more explanatory definition of stereoscopy would be:

> A general term for any arrangement of Left Hand (LH) and Right Hand (RH) views which produces a three-dimensional result, which may consist of:
>
> 1. a side-by-side or over-and-under pair of images,
> 2. superimposed images projected onto a screen,
> 3. a colour-coded composite (anaglyph),
> 4. lenticular images,
> 5. a vectograph or
> 6. in film or video, alternate projected LH and RH images which fuse by means of the persistence of vision.
>
> (STEREOSCOPY.COM, 2014)

There are many different techniques for displaying stereoscopic images, but the method for creating the stereoscopic effect can only be created by using a "toe-in" or "skewed frustum" methodology (Figures 3.2 3.3 3.4).

The **toe-in** method of creating stereoscopic images is created by projecting the left and right eye images at converging angles so that they overlap on the screen (Figure 3.2). This is recorded using

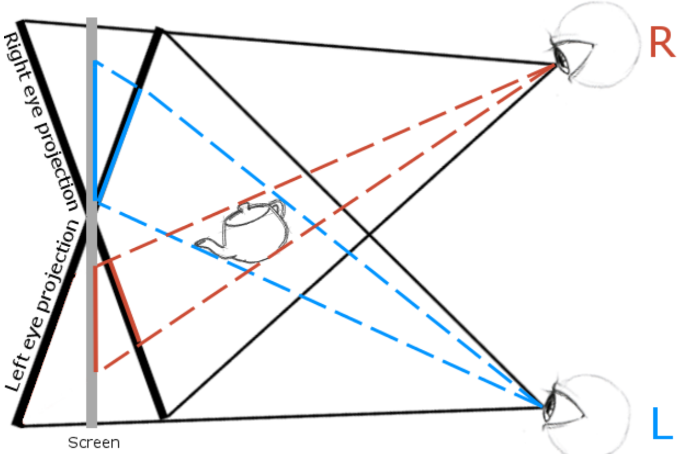


Figure 3.2: Representation of “toe-in” stereoscopy

a camera for each eye; the lines of sight of both converging at the required focal plane. The ‘toe-in’ angle refers to the angle that is created between the camera’s normal projection and that of the screen they are projecting on. This method creates a singular ‘sweet spot’ at the centre of the screen on the focal plane, where the images correctly align. The repercussions of this are that the further a user looks from this ‘sweet spot’, the more out of line the images are with their counterpart, and the more the image is stretched across the screen. This effect is called keystoning, and can be responsible for the break down of presence, or the collapse of the Plausibility Illusion. The greater the ‘toe-in’ angle is, the more pronounced and severe this effect becomes. This method is used in modern 3D films in the cinema, but for the above reasons it is not often used for 3D in VEs.

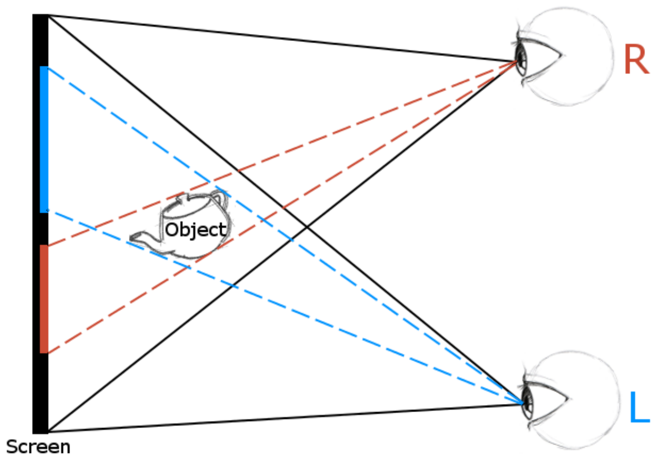


Figure 3.3: Representation of the skew on projected images that can be used to create “skewed-frustrum” stereoscopy

The **skewed-frustum** technique avoids the issues of the sweet spot created by ‘toe-in’, by projecting both left and right eye images parallel to each other 3.3. Effectively the ‘toe-in’ angle is zero. The creation of these images poses certain challenges; although the cameras are now parallel, the centre of frame for both cameras needs to be the same. This is achieved by skewing the frustum, that is, shifting the frames captured by the cameras towards each other until they overlap. In turn, this aligns the centre of frame of both cameras. This is possible in real life by using specialised camera lenses that ‘throw’ the camera frame off in the required direction. The same effect can be gained in VEs by making adjustments in the graphics render engine. A second method of creating skewed frustum is to record with a wider frame than is required and, once the images are correctly overlaid, remove the remaining areas of the frame on the left and right that do not overlap with their partner image.

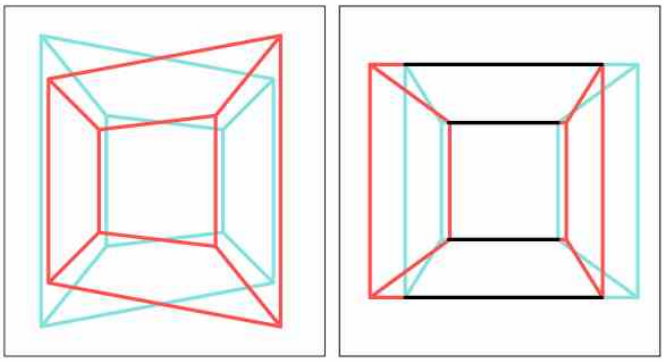


Figure 3.4: The left stereo image shows the keystoning effect caused by toe-in stereo. A viewer will not be able to merge these two views into a single 3D object. The right stereo image shows the correct result of using skewed-frustum stereo. (KREYLOS, 2014)

Other depth indicators

Stereo vision is only a part of the required solution to create a feeling of presence, or even allow a user to feel comfortable in the environment. When creating a VE, a variety of other cues need to be considered. To achieve the illusion of 3D, vector graphics are used to create isometric, single point and two point projections on flat screens. The technology is a transfer from techniques used on paper to show objects in 3D, and would not fool the mind into believing it is looking at a real 3D object. To do this we need to send a separate image to each eye using stereoscopy.

3D rendered environments use the same principles that are used in isometric or point projection drawings to give the perception of depth and distance, these are:

- **P**erspective lines: These are the lines that converge at a distance. For example the way the horizontal bars on a fence seem to get closer together the further away they are, or the way the top of a sky scraper looks a lot thinner than the bottom when viewed from below. This is not usually an accurate representation, as the fence horizontals do not actually get closer together any more than a sky scraper narrows to a near point.
- **O**cclusion: This is where a near object will block the line of sight of an object further away, effectively behind the nearer object. This is a naturally occurring phenomena in the real world, as solid objects block the progression of photons from objects behind them. Creating

a computer algorithm that could perform the same calculations was a significant challenge when computer graphics started out.

- **S**ize of known objects : Human beings are acutely aware of how big common objects like cars or other people are, therefore, by diminishing the size of known objects, the illusion that they are further away can be given. This effect is similar to perspective lines.
- **D**etails : The further away an object is, the less detail we can see in it. This is due to our eyes ability to discern the distances between light sources. The further away an object is, the smaller the arc angle that the eye observes it within is.
- **C**olour saturation : The colour of objects seems to fade out the further away they are. This is caused by the same artefact as the reduction in detail. In effect, the eye is receiving a greater mix of photons from the object through the same angle, causing a greying out. In light, the more colours that are mixed, the closer to white the user observes.
- **L**ighting and Shadows : The distance and intensity of a shadow, as it falls from an object, and its nearest light sources also feed into our comprehension of depth in a scene. This works by feeding into our past experiences in the real world of how lighting creates shadows and what we can understand about the object creating the shadow from this. For example, how thin or thick, short or tall, opaque or transparent an object is affects the type of shadow it casts.

Two important ways human beings use to perceive depth and distance are not possible using solely isometric or point projections, as these projections only deliver single fixed images.

- **R**elative motion : While travelling on a train the objects in the distance seem to be going much slower than the ones rushing by right next to the window. This relative motion indicates something about the distances and therefore depth of the scene. While a lot more obvious on a train, this affect is observed at all speeds. This parallax affect allows the user to see behind an object, effectively reducing the occlusion it is causing, by simply moving ones head. It should be noted that although relative motion can be achieved by using two static images, this change in occlusion is not possible using fixed images.
- **B**inocular distance : The distance between the user's eyes creates a disparity between the two images the user's brain interprets, allowing the brain to reconstruct a more accurate idea of the depth and distances. This is the basis of stereoscopy, and to recreate this on a piece of paper or flat screen, separate images need to be created for each eye. This distance is often referred to as the inter-axial distance.

3.2.2 Immersion

> "Absorption in some condition, action, interest, etc." (OED, 2014)

Immersion is often a poorly defined concept (BROWN et al., 2004), with descriptions ranging from how involved a person feels with their surroundings, to how much belief of the narrative and/or environment they possess. In fact, the break in the narrative from what is expected, is similar to what is described in films as the 'suspension of disbelief'. This experience is broken down into three separate levels of engagement by Douglas et al. (DOUGLAS et al., 2000) of engagement, immersion and flow, which is the "process of total involvement with life" (CZSENTMIHALYI, 1990). The greater the disruption of the narrative, that is, the more the experiences diverge from what someone accepts in real life, the further from 'flow' and 'immersion' the user will feel. To experience flow in a VE would be impracticable as the computer power required to accurately simulate the real world would be immense. With this in mind the most that can be expected from VEs and other environments is a level of immersion. In this way;

> "It should be noted that the level of immersion is completely determined by the physical properties of the system" (SLATER, 2009)

The properties of the system determine what is possible and what is not possible to simulate. For example, the resolution dictates how far one can zoom in on an object, which is information required as a user lifts an object to their face. The level of immersion could be dramatically reduced if it were to become more pixelated as it came closer. The clipping distance is another example, as it affects how far and how close, a user can see within the VE.
Within the discipline of VR the definition of immersion is defined more clearly as the;

> "extent to which a person's cognitive and perceptual systems are tricked into believing they are somewhere other than their physical location" (PATRICK et al., 2000).

So immersion is when a user is both cognitively and perceptually engaged, so that the mind perceives the alternate reality as coherent. Slater defines two distinct factors within immersion:

Place Illusion (PI) or 'Presence' is how convinced the user's perception is by external stimuli, such as vision, sound, smell, taste, and touch. PI is also affected by how close to the real world the stimuli are, for example improper head tracking would reduce the PI as the world would move in a different manner within the VE than to what the user has been accustomed to their whole life in the real world. The same could be said if a strawberry were to taste like a pear. PI is essentially the sense of existing within an environment, the feeling of *'I am here'*.

> "PI can be different in different modalities [visual, auditory...] . (...) Provided there are not inconsistencies between these modalities, there can be PI in one domain without there being PI in another. This point is important because we regard PI not as a cognitive but as a perceptual phenomenon. (...) When PI breaks, it can quickly recover." (SLATER, 2009)

Slater further separates systems into those that require further mental cognition to convince oneself of the feeling of being there, such as desktop environments, and first order systems. Here he is referring to systems that require no suspension of disbelief but instead captivate the user, without the user having to will themselves to believe.

> "In the case of a desktop system the situation is quite different, the feeling reported as 'being there' if it comes at all is after much greater exposure, requires deliberate attention, and is not automatic – it is not simply a function of how the perceptual system normally works, but is something that essentially needs to be learned (...) PI may still be reported, but this is as a consequence of additional creative mental processing. It does not refer to the same qualia as for the first order systems" (SLATER, 2009)

Plausibility Illusion (PsI) is the degree to which the user's cognition has been fooled by the VE. It is how much the user actually believes the VE is the real world, and is affected by how the user interacts, and is interacted with, within the VE. An interaction which is akin to real life will create a level of credibility in the environment with the user. An example could be that in a VE with high PsI if someone were to walk a plank over a deep gully, as they look down their heart rate would increase. PsI does not necessarily require realism, as the user could be walking along a plank made of glass over a gully with a herd of unicorns racing through it. So long as the stimulus and response are correct within the VE that the user exists within, then the PsI is also in existence.

> "PsI is therefore an illusion akin to PI – one that occurs as an immediate feeling, produced by some fundamental evaluation by the brain of one's current circumstances – 'is this real?' Of course, at a higher cognitive level participants know that nothing is 'really' happening, and they can consciously decide to modify their automatic behaviour accordingly" (SLATER, 2009)

So the goal of a VE is to have a sense of both high PI and PsI;

> If you are there (PI) and what appears to be happening is really happening PsI then this is happening to you! Hence you are likely to respond as if it were real. We call this 'response-as-if-real' (SLATER, 2009)

The level of immersion within a system, although determined by the capabilities of the system is also dependant upon generation of belief within the user that they are actually in the VE. However PI and PsI are not inclusive; a system can create an environment with all the correct external stimuli, but if what is simulated in the environment does not appear correct to the user then PsI will not be maintained. An example of this is natural movement; a great deal of time and effort has been spent on motion capture so that characters move correctly within computer environments. Although subtle, it has a great deal of effect on the PsI. A user watching people walk like lego men would experience PI without PsI. While in contrast, watching a dinosaur with natural movement taken from motion capture of another animal, one would be more likely to experience PsI. However if the display resolution was poor, or there was almost no sound, then they would be without PI. To gain a high level of immersion, the combination of both PI and PsI are required.

A key part of VR is the ability to interact with the environment. When this is possible, the illusions of place and plausibility are increased proportionally to the verisimilitude of the VE. The closer to reality the VE is then the greater the illusion, in this respect adding feedback for additional senses would increase the verisimilitude of the VE.

3.2.3 Haptics

> "Technology that provides a user interface based on the stimulation of the senses of touch and movement (kinaesthesia); the branch of science or engineering concerned with such technology" (OED, 2014)

Haptics allow the user to virtually touch, feel and manipulate objects within a MR environment. Similar to computer graphics, haptics serve as a link between the user and the MR environment. To achieve this purpose a haptic system requires two loops (Figure 3.5, one which takes place within the user themselves, and the other within the computer system that is creating the MR.

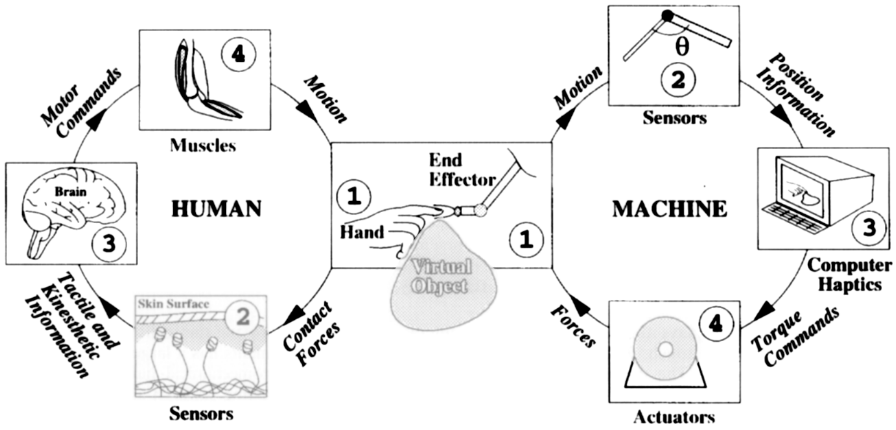


Figure 3.5: Haptic interaction loops between user and machine (M. A. SRINIVASAN et al., 1997)

The human side of the loop is the feedback mechanism that takes place in the real world. A human moves their manipulator (limb) and a part of their skin comes into contact with a solid object. Electrical impulses are then transmitted from the point of contact to the afferent neural network, of which the brain is a part. The magnitude of the impulses correspond to the resistance to force, the object has expressed. The brain will then interpret this data and transmit electrical impulses through the afferent neural network to the muscles which will in turn move according to the decided actions. Data from the stimulated sensors is inverted and interpreted within the brain, with the exception of reflex actions. This information is comparable to that of tactile information gathered from the skin in contact with an object. A second set of information is also gathered from a similar but continuous looping mechanism, known as kinaesthetic or proprioceptive information relating to the positions of the limbs.

The computer or haptic device side of the loop works in a similar way to the human mechanism. The user triggers sensors within the device by making contact with it. These sensors are also either of a tactile or kinaesthetic nature. Tactile sensors could include pressure sensors, heat sensors, and electric impulse sensors, whereas kinaesthetic sensors would measure the movement of the device, so could be accelerometers, motors, variable resistors, and any form of tracking. The signals from these sensors are then relayed to a computer which interprets them. The computer then uses the models of objects to calculate the real-time feedback in the form of torque to the device's actuators or as an electrical impulse to the tactile pads. This, in turn, leads to the user experiencing a tactile perception of the virtual objects.

Once such example of such a system is the welding simulators. Welding would seem to be an ideal application for VR and haptic feedback simulations. Such a system would allow for users to train without the safety overheads and cost of experienced training personnel. It could also allow for a wide range of simulated welding situations and techniques. Prior work was undertaken at Rolls-Royce Civil Nuclear Manufacturing to analyse existing technology on the market for welding simulation. A list of options was broken down by system features and benefits, and the Lincoln Electric Welding simulator was chosen as the most viable candidate at the time of research. A cost benefit analysis of the Lincoln Electric welding simulators was done to see if it would be worth installing the systems into the apprentice training centres. While the results seemed to indicate that with enough apprentices taking part in the program that the systems would cost less than the current training systems, a number of drawbacks were identified. The greatest of these was the lack of real haptic feedback within the simulator. Through in depth discussions with the Rolls-Royce welding teams, it transpired that the feel of the welding process was deemed as one of the most important outcomes of training. Furthermore the welding team felt that simulators would also fail to produce the same heat, spark, and hot metal feedback that is produced when conducting welding in the real world. Finally a lack of a noisy factory floor background and requirements to work in sub optimal settings would hamper the application of any weld training done in the unlimited space and carefully controlled sound environment within a simulation. A significant aspect raised by welders in the discussion, was the argument of muscle memory generated from repeated welding. One welder went so far as to claim that he would often dream a weld, and he doubted that any simulation could produce that effect.

Although haptics is still a developing field of research, the addition of kinaesthetics or tactile feedback can greatly enhance the user's understanding of a system. Imagine a user learning to play tennis on a system that had no feedback from the collision of ball and racket. When they were exposed to tennis in the real world, it would be a shock, and require some adaptation on their part. When applying MR to training scenarios, this area needs to be understood and taken into consideration as there maybe little point in training someone to virtually hammer a nail.

3.3 Data, Information, Knowledge and Wisdom

Less haptic feedback might be required when users combine data sets and use visual information to create an integrated single environment. The next section elaborates on what is meant by Data, Information, Knowledge, and Wisdom. As previously mentioned, these are seen as an integral way of understanding one of the best use cases of MR; the combination of data sets and information in a format that is quickly and easily understood by a wide group of people. In this way, it is hoped that MR can be used to help users gain access to knowledge, allowing them to make more informed decisions.

It is now widely accepted in academia that information is part of a hierarchy which includes data, knowledge and wisdom, The data–information–knowledge–wisdom hierarchy (DIKW) or more simply the wisdom hierarchy is a concept that was first established by Russell Ackoff in 1989 (ACKOFF, 1989). Although Ackoff is widely credited with the creation of the hierarchy, the first mention of a similar but more rudamentay idea is by T. S. Elliot in *The Rock, 1943*

> "Where is the wisdom we have lost in knowledge? Where is the knowledge we have

> lost in information?"

published in 1943, it can therefore be assumed that the idea had been in existence before being published by Ackoff.

Ackoff includes understanding within the hierarchy, but this has been disputed more recently and is instead now referred to as the method of transition between the categories (BELLINGER et al., 2004). Also included within Ackoffs' paper is intelligence, however it exists outside of the hierarchy. Within all forms of the DIKW each succeeding category contains the previous one, so that there can be no wisdom without knowledge, no knowledge without information, and no information without data. Ackoff defines the members of the categories and their methods of transition as:

> **Data** are defined as symbols that represent properties of objects, events and their environment. They are the products of observation, but are of no use until they are in a usable (i.e. relevant) form. The difference between data and information is functional, not structural.
>
> **Information** is contained in descriptions, answers to questions that begin with such words as who, what, when and how many. Information systems generate, store, retrieve and process data. Information is inferred from data.
>
> **Knowledge** is know-how, and is what makes possible the transformation of information into instructions. Knowledge can be obtained either by transmission from another who has it, by instruction, or by extracting it from experience.
>
> **Wisdom** is the ability to increase effectiveness. Wisdom adds value, which requires the mental function that we call judgement. The ethical and aesthetic values that this implies are inherent to the actor and are unique and personal.
>
> **Intelligence** is the ability to increase efficiency.
> (ROWLEY, 2007)

The first three of these all pertain to the past, with only wisdom applying to the future as it aids judgement. To gain wisdom someone must pass through the other stages, by processing the raw data, analysing it, and using it to answer questions. This information must then be processed further either by others or by experiencing the same information several times. The result of this experience can be considered as knowledge, that is, the summing up of information. The final step to wisdom can be attained using that knowledge to add value to decision making processes, or to increase the effectiveness of the knowledge.

To be explicit: Data is a fragment of information, a value which holds no relation to any other, such as:

It is wet

Information is a series of data points that hold a relationship to each other

The wet cat is black

Knowledge is like a set of information that can be interpreted to lead to a conclusion

It is raining, rain is wet, the cat is in the rain, so the cat is wet

Wisdom is an understanding of the interconnectivity of information

Rain is part of the known water cycle, driven partly by the sun, which the earth orbits....

The flow from data to wisdom is particularly useful when demonstrating the use of computer programs, as it can be said that the further up the pyramid the computer can convert the data, the higher the value and meaning that it has created for the user (CHAFFEY et al., 2010) (Figure 3.6). However, computers are a lot more adept at working with data than they are at demonstrating intelligence or wisdom. Therefore the higher up the pyramid, the less useful a computer program can be said to be.

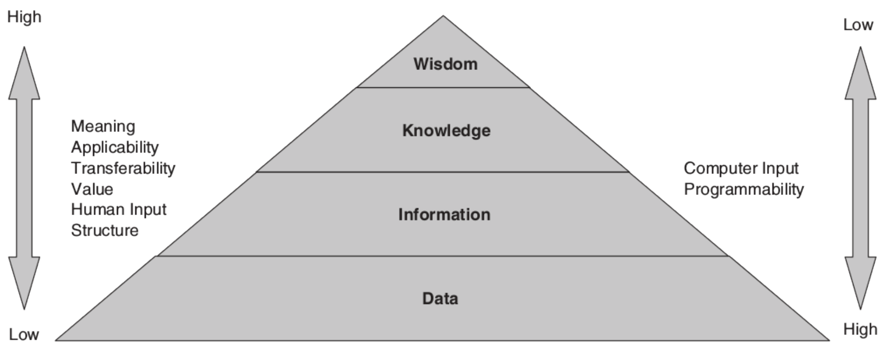


Figure 3.6: The DIKW hierarchy expressed as a pyramid (ROWLEY, 2007)

Computer programs are well known as data analytic tools and, through the use of correlation diagrams such as graphs, help people to convey information to each other. In this way, data sets can be combined to show relationships, which is information, as defined above. Similarly, a combination of different sets of information can help understanding, which leads to knowledge. MR can be used to take abstract data and information, combine it in a single location, and show it to users or groups in a VE that demonstrates how they fit together to assist in the formation of knowledge. For example, the VE could be an approximation of the real world where the data was gathered. This could be of use where large numbers are concerned, a group of 1000 people compared to a group of 10,000 could be difficult to visualise. However if shown in a VE they would be easier to compare, and consider as people. If the two groups were then placed near a known reference, say a football field (virtual or real), then we are presented with context, and understanding could lead to knowledge. That is a genuine knowledge of how many people there are, based on comprehension developed in the real world.

The conversion of data to information takes place regularly within industry, the next section will highlight some scenarios from manufacturing that would be considered as good targets for integration into MR environments.

3.4 Data & information in manufacturing

The world is now full of devices capturing data all around us. This is equally true in the world of manufacturing, where devices are used to connect initial design phase data and information to final manufacture and Product Life-cycle Management (PLM) analysis. The transition of the data sources into information by staff, and development of simulations can aid in the decision making processes. The retained information also allows for things too small, too large, or relationships from data too disparate to be identified. Below is a brief overview of sources of data within manufacturing:

3.4.1 Data capture

Coordinate Measuring Machine (CMM) is a more precise method for measuring the geo-spatial coordinates of a product. By using a touch tool tip, a CMM takes repeated measurements of the product. These are currently used to check conformance and that the parts meet the required quality and tolerances. However they can also be used to build 3D maps of existing parts.

Dye Penetrant Inspection (DPI) or Liquid Penetrant Inspection (LPI) involves painting a dye over a product, which then accumulates within cracks. Excess dye is removed, and a developer or UV light applied to make the dye more visible. In this way Dye Penetrant Inspection (DPI) helps identify surface cracks which could otherwise be hard to see. Again as the final inspection is visual then digital images must be taken for the data to be used in future calculations.

Eddy current testing & inspection uses electro magnets to induce an eddy current on the surface and sub surface of metals. The deviations of the measured currents from expected values indicates abnormalities in the make up of the metal. These can be interpreted as areas of non-conformance such as cracks, or a build up of undesirable materials within the metal. Similar to Ultrasonic Testing, geo-spatial data is also required for computational assessment of Eddy currents to be of use.

Infrared sensors can be used to analyse the heat distribution within a building. This can identify electric cables, areas of intense energy use or areas where it is unsafe for people to work. Tool and machine heat distribution can be used to analyse the performance and be used to indicate when end of life could be approaching. In heat treatment processes, the distribution of the heat is of particular interest.

Laser scanning works using Time of Flight (ToF) to calculate the distance from an emitter to the object a laser has reflected off. This gives a location of the object, relative to the position of the scanner. Modern laser scanners can combine the scanned data with still images taken from a camera to assign RGB or CMYK color values to the points, resulting in a coloured point cloud data set.

Machine operational parameters could include data such as; operating time, Overall Equipment Effectiveness (OEE), coolant levels, cycle time, number of operations performed, or still to perform, tool paths, weight of swarf, etc. All of these could be digitised and assembled over networks like ORiN (ORiN, 2014).

Physical measurement of the geometry and geo-spatial coordinates of all the components can be done by direct measuring. Which in turn would need to be entered manually into a computer for use in calculations.

Radiographic Testing (RT) is a once through technique that exposes a part with a photographic plate behind it, to electromagnetic radiation. Electromagnetic rays which are not blocked will be registered on the plate. In this manner areas of high density will show up less exposed on the plate, and so lighter. Digital plates can be used to allow direct transfer and interpretation by computer algorithms, but this technology is yet to be adopted by the nuclear inspection community due to the level of interrogation still being in its infancy.

Thermo-couples provide a similar function to infra-red sensors but need to be placed in physical contact with what they are measuring. The advantage of thermocouples is that they are cheap, and once mounted the location can be stored manually.

Ultrasonic Testing (UT) scanning is done by running detectors back and forth across components to create a 2D or 3D map of the internal structure of the material. Unless this information is combined with geo-spatial data then although the information could be analysed by a computer, any defects found and their location would be unreferenced.

Visual inspection done by qualified or experienced operators helps to identify challenges and/or defects both on the shop floor, and on the products and parts themselves. Data gathered in this way is qualitative, not quantitative so is difficult to translate across into a computer environment or simulation. However cameras can be used to visually inspect by proxy, in which case, video feeds and/or still images can be imported into a computational environment for analysis.

For data to be used within a VE it must be converted into a quantitative format: in many cases this is done by a process of digitisation. Often digitisation alone will not suffice, and configurations or algorithms must be used to retrieve the useful information from the data set. This is a substitute process for skilled operators, e.g. a Non-Destructive Testing (NDT) specialist could have trained for years to identify specific defects found through ultra sound scanning. Once this information is digitised a computer could scan through and identify these defects. The computer would be able to scan more data, in shorter periods of time than the specialist could hope to achieve. Within businesses this can lead to a reduction in costs. This is covered in more detail in Section 4.2.2.

The combination of data sets can lead to the creation of information that can be easily interpreted by people (Figure 3.7). The form of information that humans are most used to analysing is that which surrounds them everyday. The closer virtual data is to the real, the more adept the user is at interpreting that information. A combination of information, set in an environment that closely resembles the real, such as a VE, can therefore be said to hold value.

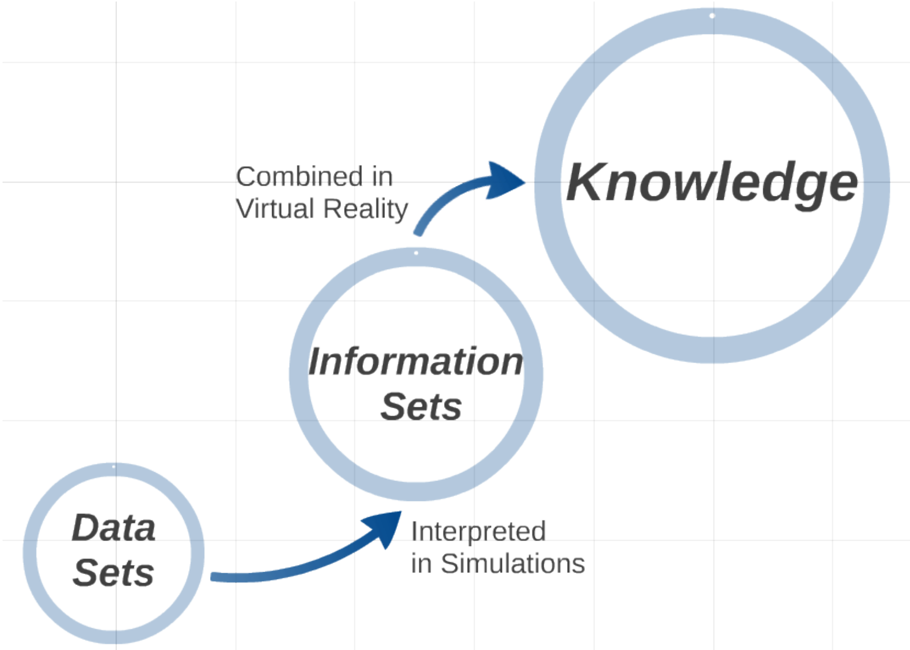


Figure 3.7: Proposed transition method for improving understanding and therefore gaining knowledge of data sets.

Various methods of displaying data sets and helping transform them into information are given in the next section. Examples are used that are both relevant to manufacturing techniques and have previously been imported into VEs.

3.4.2 Data interpretation

Chemical interactions are modelled using variations on physics engines, which deal with chemical laws such as cross section and absorption rates. They are widely used in pharmacy and in industrial chemical companies. One of the pioneering uses of VR was for the pharmaceutical industry (BATTER et al., 1972), and OpenSource software is now available to simulate molecules in VR (SCHRÖDINGER, 2014)

Computer Aided Design (CAD) uses geo-spatial data to construct a computer graphic representation of a product, or parts of one. The model can be derived or created during the design process, or captured from an existing part/product using methods outlined in section 4.2.2.

Computational Fluid Dynamics (CFD) uses numerical methods to represent the flow of liquids, predominantly using the Navier-Stokes equations. Computational Fluid Dynamics (CFD) is used throughout industry and requires data sets including temperature, viscosity, pressure, density, liquid to gas ratio, and velocity of the mediums being calculated. It is used to calculate drag and other forces resultant from fluid flow around an object. The transfer of CFD into VR has been accomplished by both NASA and Boeing (ZIMMERMAN et al., 1987), and is currently used in chemical engineering (X. LIU et al., 2012)

Finite Element Analysis (FEA) is an iterative calculation similar to CFD which relies upon meshing, and the interrelation of calculation cells. Stated as a numerical method for finding the boundary conditions, it is most applied in industry to determine the build up of stress in materials of given geometry. This is done to evaluate where a fracture, or break may occur, and how likely it is. Finite Element Analysis (FEA) is used in VR for suturing and other surgical simulators (MARESCAUX, CLÉMENT et al., 1998) and although it has been applied elsewhere, it is yet to become common practise (LIVERANI et al., 1999).

Heat transfer equations combine to model the distribution of heat, which is of vital importance in fields such as thermal hydraulics. This in turn is relevant to civil nuclear reactor technology. Methods have been developed to model the transfer of heat for a more advanced form of haptics (BENALI-KHOUDJA et al., 2003). It is used with CFD in atmospheric calculations in VR and the chemical industry (X. LIU et al., 2012; RATTS et al., 2000)

Materials control is used to assess the levels of inventory and materials on the shop floor. This allows Material Resource Planning Controllers (MRPCs) to ensure that the correct flow of materials and resources are always available to staff. Boeing was the first to use materials control data within a VE (H. A. SCOTT, 1994)

Human factors are the combination of data on people: whether how far someone can reach, or the distribution of heights of people etc. It is used to assess not only if suggested operations are possible, but how difficult or how much strain an individual may be under while performing. One of the main areas that human factors are applied to is Health Safety & Environment (HS&E). Human factors are often considered in VE while working on assembly design and development (RASHID et al., 2012; H. A. SCOTT, 1994)

Particle transport models such as MCNP, are used to predict radiation damage, chemical infusion, oceanic transport etc. The data required is as varied as the cases used for the model. The distinguishing feature of the model is that work is calculated in quanta. If a single particle acts in this matter, then a system of them would interact in this fashion. Monte Carlo simulation for the calculation of dose rates in workers has been applied in VIP-man, a virtual human (XG XU et al., 2000),

Physics engines are used to create approximate simulations of physical systems, such as gravity within the solar system, or rigid bodies on a car frame. The types of data that are used in the simulations depends on what system is being modelled and the outcomes desired. Physics engines are now regularly provided within game engines and are available in VEs (JACOBSON and LEWIS, 2005)

Routing is a description of the paths that parts take through a factory. It is assembled from the locations of where processes are required to take place, and the routes that a product can possibly take between them. This could range from a local scale e.g. rubber conveyor belt to a machine moved within a few metres, to a global scale e.g. an export to a Japanese forging company. Routing in VR has been explored for assembly (ABDULLAH et al., 2003), and is considered alongside Discrete Event Simulation (DES) in other studies (DANGELMAIER et al., 2005)

Scheduling is a way of combining the known data of process times and sequences in a way that a schedule for delivery is framed. This is important at all levels of manufacturing. The data sources can relate directly to that gathered on the shop floor. Scheduling has been used in VR simulations of a multitude of factories, and is a developing area of research (J. J. KELSICK et al., 1998; NANCE, 1993)

3.5 Simulations

> "The technique of imitating the behaviour of some situation or process (whether economic, military, mechanical, etc.) by means of a suitably analogous situation or apparatus, esp. for the purpose of study or personnel training." (OED, 2014)

There are three main types of computer simulation:

Discrete Event Simulation utilises a mathematical/logical model of a physical system that portrays state changes at precise points in simulated time. Both the nature of the state change and the time at which the change occurs require precise description. Customers waiting for service, the management of parts inventory or military combat are typical domains of discrete event simulation.

Continuous simulation uses models created from sets of equations, often of physical systems, which do not portray precise time and state relationships that result in discontinuities. The objective of studies using such models do not require the explicit representation of state and time relationships. Examples of such systems are found in ecological modelling, ballistic re-entry, or large scale economic models.

Monte Carlo simulation , the name given by John van Neumann and Stanislaw M.Ulam to reflect its gambling similarity, utilises models of uncertainty where representation of time is unnecessary. The term originally attributed to "a situation in which a difficult non-probabilistic problem is solved through the invention of a stochastic process that satisfies the relations of the deterministic problem". A more recent characterisation is that Monte Carlo is the method of repetitive trials. Typical of Monte Carlo simulation is the approximation of a definite integral by circumscribing the region with a known geometric shape, then generating random points to estimate the area of the region through the proportion of points falling within the region boundaries. (NANCE, 1993)

There are a great number of simulation programs for converting data sets into information for consumption. As will be highlighted in Section 4 one of the major problems is converting the data from proprietary formats into formats that can be combined with others in MR. Some software comes with either Application Programming Interfaces (APIs), or Software Development Kits (SDKs) allowing programmers to get them to “talk” to other software or hardware systems. Without such an interface, researchers are often left to use or develop open source alternatives to allow the conglomeration of data and information into a single environment. The following chapter will provide several examples of both the use of open source APIs and SDKs as means of integrating services into MR.

3.6 Summary

This section has detailed a number of key definitions, and provided some explanation to concepts vital to the understanding of what a good VE training simulator should provide. The use of VE techniques such as stereoscopy, can increase the levels of user immersion, and so increase the useful transfer of skills developed in a simulation into the real world. The chapter then talks about the transfer of data into information, and how a context-sensitive environment such as a VE can aid users in that conversion of data into information. The chapter concludes with examples of data sources that have been imported into VEs to achieve this, before it finishes with simulations. Which in this context can be thought of as a systems of processing data into information prior to use within VE. In such cases the VE is used to assist the users in turning the information into predicted knowledge.

3.6.1 DAAC

The Digitally Assisted Assembly Cell (DAAC) uses most of the techniques discussed in the early sections of the chapter. The VE is accessed using the HTC VIVE which has a skewed frustrum stereoscopic display. For users outside of the VE, objects have been added to the VE to give users a good idea of the relative size of objects. This allows users to gauge how far away one object us from another even without a stereoscopic display. The entire VE is a facsimile of the real manufacturing cell, and the high levels of verisimilitude serve to increase both the PsI and the PI for users. This is helpful for users when transferring information and insights gained within the VE into the real manufacturing cell. The experience of PsI and PI within the Digitally Assisted Assembly Cell (DAAC) must be a step improvement from previous implementations of VE for training in manufacture. The use of game engines ensures that users comparing the VE to their game titles at home, experience less blow back and have a higher level of acceptance for the technology. The use of commercially available software and hardware also helps to ensure that the methods of interaction, visual cues, game trope actions are all inline with what users could be experiencing at home. This makes the DAAC easier to pick up and use straight away, reducing the training times required for new users.

The implications for training users to both use the tool, but also in manufacturing processes is quite clear. Perhaps less clear is the benefits of using the VE as knowledge retention and transfer tool. The DAAC can be seen as an active system that can be used to instruct users, but also importantly to store data.

The VE can be used to import Computer Aided Design (CAD) and store the relative locations to each other. This could be particularly useful for storing cell layouts, as the different layouts could be switched between depending on the task that was to be undertaken within the real cell. User in the real cell could see the virtual cell through the Android tablets and change the cell accordingly. Furthermore items could be set to be projected out off the virtual environment and into the real to assist in placement of product or machinery. Objects or machines that the CAD either could not be imported due to file formats, complexity or having no existing CAD can be laser scanned, and imported as a point cloud mesh instead.

The data stored within the DAAC system need not be limited to geo-spatial data, about the size and shape of objects. The game engine allows almost any amount of meta data to be attached to the objects. This could be anything from delivery times, to material type and constitution, what processes can or can not be performed on the object. Although the DAAC does not import or use meta data, there is no hurdle to it doing so. What the DAAC does store is work instructions, which can be used to store any number of processes and controls, furthermore they can be used to advise and guide the workers both in training, and during actual manufacturing. The work instructions can be easily designed and tested within the VE, they can even be used to help assist trainers in the VE when working with staff in the real manufacturing cell.

A third and perhaps final area that is stored within the virtual cell is the human tracking data. The movements of users within a manufacturing cell can be recorded and replayed in the VE. This allows for users to observe the manufacturers movements within the VE at any future point in time and space. This will allow for the storage of both explicit and some implicit data on how manufacturing techniques are carried out. This could prove useful for demonstrating how long a task takes to carry out, and what volumes within the cell need to be clear to allow for that. This data can then be exported into DES to improve their accuracy. It also allows for the storage of manufacturing process for future generations to review.

Through the importation of data into the system, it start to become information, this allows users to interact with it and convert it further into knowledge.

CHAPTER 4

Literature review

This literature review defines the key areas of Mixed Reality (MR) as the Virtual Environment (Section 4.1), Presence or Place Illusion (Section 4.1.1), sensory feedback (Section 4.1.2) and interactivity (Section 4.1.3). Using the definitions, the review then describes the narrative of MR and its uses through history (Section 4.1.4). The history of MR is one that progresses from a flight simulator made of old organ parts, to the birth of computer graphics, then onto diversification into uses in military, medicine and architecture and finally its failure to deliver on commercial promises in the 1990s. The section finishes with a discussion of the current renaissance of interest in the field and the uptake of the technology by enthusiasts and major companies alike.

The academic literature covering the use of MR in design, manufacture and in-service is summarised in the subsequent sections 4.2.1. The review concludes with discussions pertaining to gaps in the literature and challenges still to be faced (also Sections 4.2.5).

From social networking (DAMER et al., 1997; MESSINGER et al., 2009) to military simulations (EISLER, TYRA et al., 2001) Virtual Reality (VR) is being used throughout society. With the increasing prevalence of the technology, the definition of VR can be obscured behind marketing hype, misuse and miscomprehensions.

> "*Virtual reality* is composed of an interactive computer simulation, which senses the user's state and operation and replaces or augments sensory feedback information to one or more senses in a way that the user gets a sense of being immersed in the simulation (virtual environment). We can thus identify four basic elements of virtual reality; the virtual environment; virtual presence; sensory feedback (as a response to the user's actions) and interactivity" (SHERMAN et al., 2003)

4.1 Virtual Environment

The Virtual Environment (VE) describes Computer Generated (CG) objects and their respective inter relationships. VR is the observation of a VE, importantly for VR it must also allow the user to participate or interact with it. There are similarities with John Wheelers 'Participatory Anthropic Principles' theory in quantum mechanics (WHEELER, 1975), in which participation with the world defines it. CG objects can be created to look, sound, feel, smell and taste the same as real world counterparts but they will not be created on screen unless we are looking at them in the VE. Similarly they will not be heard, tasted or smelled, unless we interact with them in the VE. A VE contains the information for the CG objects, this includes their geometry, spatial coordinates and the rules that govern their interrelationships. These could be; if/how they collide; what physics equations govern their movement and what they do when the user interacts with them etc.. The VE contains the mathematical relationships that create a world that mimics our own, or the one desired.

4.1.1 Virtual presence

Presence in VR is the feeling of 'being' in the VE rather than the real environment connected to a display system. Douglas et al. have separated presence into engagement, immersion and flow (DOUGLAS et al., 2000). Flow is unlikely to be fully achievable in a VE as any user in a VE also exists within real reality (CZSENTMIHALYI, 1990), however immersion and engagement are possible. Slater defines presence as Place Illusion (PI), which is how convinced the user's perception is by external influences. The level of presence is therefore determined completely by the physical properties of the system. Immersion and engagement can then be grouped together into Plausibility Illusion (PsI). PsI is akin to Place Illusion (PI), but relates to how close the feelings and responses to the VE are to those experienced or caused in the real world (SLATER, 2009). The opposite of PI exists in the form of absence, a notion analogous to the level of detachment felt from the narrative when reading a book (WATERWORTH et al., 2003). PI and absence are important concepts within MR as they can be used to describe the effectiveness of the VE in fooling the user. In some cases, such as training in situations of high duress, a high degree PI would be desirable. However when treating patients for phobias or Post Traumatic Stress Disorder (PTSD) a level of absence could be essential in avoiding causing psychological damage (KRIJN et al., 2004).

4.1.2 Sensory feedback

Presence or PI is created by external influences on the user, while PsI relies on correct sensory feedback. If an object dropped in a VE were to float instead of fall, then the PI would be broken, and consequently the level of immersion reduced. Knowledge of what senses will be required, if the goal to "fool the user's senses" is to be achieved (HOLLOWAY et al., 1995). The contributions to the whole for each sense is listed below and was ascertained by Morton Heilig for his work on a the cinema of the future:

> 70 % sight - 20 % hearing - 5 % smell - 4 % touch - 1 % taste (HEILIG, 1992)

The research shows why sensory feedback in VR is predominantly visual, with the recreation of realistic sound, secondary, and other senses often neglected in comparison. Directional sound was deemed important by Cruz et al. who used a surround sound system in CAVE systems since its inception (CRUZ-NEIRA, SANDIN, DEFANTI et al., 1992). More recently, efforts to improve the software of positional audio have been made (DALY et al., 2002), used within VEs (SYGNALOW, 2014), and combined with improvements in hardware (OCULUS, 2014). Olifactory devices have also been in existence and subsequently developed, since Morton Heiligs original Sensorama device (CATER, 1992). Currently iterations can be found as both head mounted and arm worn devices (MOCHIZUKI et al., 2004; YAMADA et al., 2006). Only recently has the replication of taste been made possible by Ranasinghe et al. (RANASINGHE, CHEOK et al., 2013; RANASINGHE, K. LEE et al., 2014). It remains to be seen how this will be used outside of its current scope in gaming. Kinaesthetic and tactile feedback in MR was also first introduced in Morton Hieligs Sensorama. Wind was blown at the user and their saddle vibrated. A full taxonomy of devices and their uses are discussed in Srinivasan et al. (M. A. SRINIVASAN et al., 1997), where despite Heilig attributing a contribution of only 4% to it (HEILIG, 1992), the author stresses the importance of haptics as the primary form of interaction with a VE.

4.1.3 Interactivity

The ability to interact with a VE is essential for VR, and a wide variety of devices exist to allow a user to do so. These range from trackers that map the location and/or orientation of parts of the user in the real world and transfer across into the VE (ROLLAND et al., 2001), to force feedback arms and tactile interfaces. Interaction with a VE should be performed in as close a manner as possible to the real world, to allow maximum cross over of skills from MR to the real world (TARR et al., 2002). In this way, the majority of today's haptic devices fall short, as the mechanistic approach leads to a poor approximation of real world feedback. This is especially true in tactile devices (ARAS et al., 2014); whereas kinaesthetic devices provide more convincing feedback, they tend to have a limited range of movements and allow movement only within a set volume (M. SRINIVASAN et al., 2009). Voice commands are now commonly used in VEs to communicate with other users and as a method of controlling aspects of the VE (CHU et al., 1997; MONAHAN et al., 2008; SAVAGE-CARMONA et al., 1998). Finally, a variety of ways have been developed to track where a user is looking while in the VE (DUCHOWSKI et al., 2000). Knowing where users are looking would be of particular interest during component inspection.

Having described how the literature defines VR and the its most important components, a two questions remain to be addressed. Why and for what was all the technology for MR developed?

4.1.4 History

The first mention of a VR system was most likely Aldous Huxleys 'feelies' in his book "Brave New World" (HUXLEY, 1932). The first use of a simulated environment was the Link Flight Trainers, which were commissioned by Edwin Link to train pilots on the ground (LINK JR, 1937). The simulators, although free of modern computer technology, made the case for simulated environments. They also proved the effectiveness of simulated environments, even if, as was the case with the first Link Later designs, the sensory experience was very limited. Edwin Link was the son of an organist and used organ parts and pressurised air to control an altimeter and movements of the trainer. The only other device on board was an artificial horizon. Using only two instruments,pilot training was accomplished and was considered effective enough to sell thousands. This started a lengthy history of simulated environments being both used, and developed, by the US military.

The idea of a training simulator saw more mainstream adoption in the 1950s as Aetnas Drivotrainer was used in a variety of driving schools in the UK and US. Similar to the Link Flight Trainers, the Drivotrainer was a reduced size adaption of its counterpart that tracked the users movements. The single seater contraptions were used in classrooms, where multiple simulators would watch a film projected onto a screen at the front of the class. The Drivotrainer was slightly more advanced than

Figure 4.1: Parallel parking on the Aetna Drivotrainer [at Brooklyn High School of Automotive Trades](BROOKLYN-LIBRARY, 2009)

Figure 4.2: The students' skill in meeting traffic sequences shown on motion picture screen at front of classroom is automatically recorded on a score sheet (BROOKLYN-LIBRARY, 2009)

the flight simulator as all the user's actions were relayed to a computer to assess them. Their actions were checked against the set of actions an ideal driver would take at the correct moment in the film (BROOKLYN-LIBRARY, 2009).

Inspired by Huxley's idea of a more immersive cinematic experience, rather than the dystopic future, in 1960 Morton Heilig implemented an important extension to the Drivotrainer simulators. The Sensorama was a viewing box, that provided stereo images, accompanied by sounds, smells, air flow and even vibrations through the seat (Figures 4.3 4.4)(HEILIG, 1962). Two years later Heilig patented a Head Mounted Display (HMD), complete with audio and olifactory feedback mechanisms (Figure 4.6) (HEILIG, 1960). Despite encompassing all of the user's senses, neither device provided a method of interacting with the VE, and so should not be described as the first instances of VR.

The first Head Mounted Display was created a year prior to Heilig's patent (Figures 4.6). Called Headsight, it was created by Philco engineers and used to control a remote camera. The user's head was tracked, and the corresponding movements passed by wire to the camera, creating the first instance of 'tele-presence' (COMEAU et al., 1961). Although creating the illusion of being elsewhere, the device was designed for surveillance and not the viewing of a VE. The design of the Headsight was to form one half of a solution created by Sutherland in the middle of the 1960s.

The other half of the work was conducted by General Motors in 1964 as they pioneered research into a Design Augmented by Computer (DAC) system, to assist in automotive design (JACKS, 1964). This created the final piece of the platform that Sutherland was to use in his bid to create the '*Ultimate Display*'(SUTHERLAND, 1965). He would be the driving force behind development towards VR for the next decade and is still considered one of the forefathers in computer graphics.

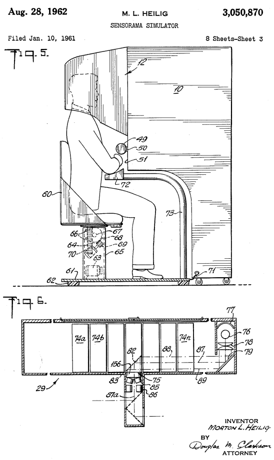


Figure 4.3: Sensorama viewing booth (HEILIG, 1962)

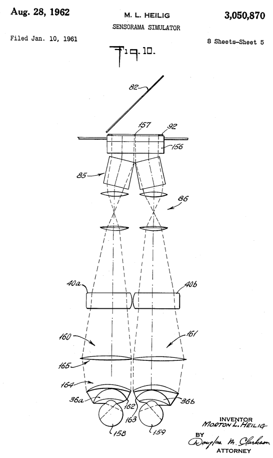


Figure 4.4: Sensorama stereoscopic ray diagram (HEILIG, 1962)

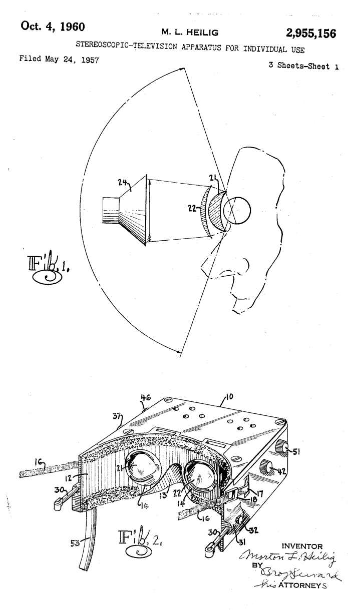


Figure 4.5: Personal stereoscopic viewer (HEILIG, 1960)

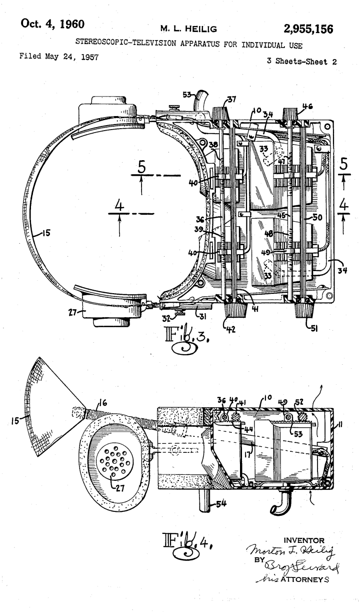


Figure 4.6: Personal stereoscopic viewer with aural and olifactory devices (HEILIG, 1960)

To achieve his goal Ivan Sutherland worked with Gary Hodgeman creating graphical display algorithms that would allow for a real time rendering process (SUTHERLAND and HODGMAN, 1974). Their work on clipping planes and vertex culling was seminal and would be used for decades to come. With a VE now possible, Sutherland created the *'Sword of Damocles'* in 1968 (Figure 4.7)(SUTHERLAND, 1968). The HMD used ultrasonic phase difference tracking to synchronise the user's movements with that of a limited VE, the results of which were projected onto transparent plates mounted in front of the users eyes (SUTHERLAND, 1968).

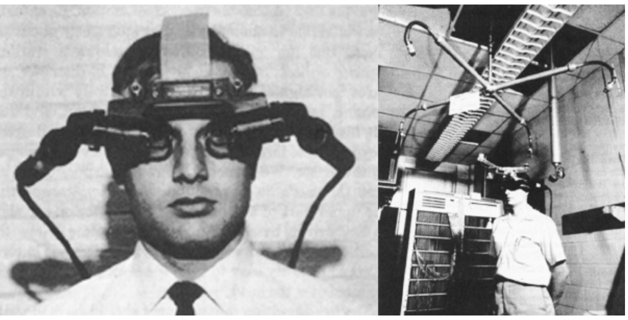


Figure 4.7: Ivan Sutherland demonstrating the 'Sword of Damocles' HMD (SUTHERLAND, 1968)

Figure 4.8: The GROPE I remote manipulator being used to test molecular interaction (BATTER et al., 1972)

With the creation of a digital stereoscopic HMD the next challenge in creating a VR was to design a more interactive environment for the user. The VE to do this was finally established in 1971 at the University of North Carolina (UNC). The aptly named *'GROPE I"* device was a mechanical force feedback arm, that served as the link between the user's arm and the VE . The movement of the user's arm was translated into movements in the VE, allowing the user to manipulate objects within the VE (Figure 4.8)(BATTER et al., 1972). The GROPE arm has since seen many iterations and funding streams. Its most common function was to allow chemists and biologists to manipulate molecules and to check fitting mechanisms at interactions sites (BROOKS JR et al., 1990).

Interaction with the VE continued to play a dominant role in research as Evans and Sutherland created the successor to the Link Laters in 1973 (SHERMAN et al., 2002). The update involved incorporating a CG virtual horizon and up to 2,000 light points onto screens seen by the pilot. The number of light points was limited by the capabilities of computer graphics available at the time. In 1977 the first wearable interaction device was created: the Sayre glove used a variation in light intensity down fibre optics strapped to a glove to determine the movements of the user's hands. This could then be translated across into a VE and when combined with a tracking device, provide the user with manipulation controls with a higher resemblance to reality (STURMAN et al., 1994). It would be another five years before any further advances in interaction technology were made.

In the interim Eric Howlett created the Large Expanse Enhanced Perspective (LEEP) optics system in 1979, which had a vastly improved Field of View (FoV) for HMDs at the time, and consequently would be integral to all HMDs from then on(Lau et al., 2013). The driving force for the LEEP system was the recognition that the small 45 degree FoV that had been available, led to significant reductions in the feeling of immersion (Howlett, 1990). This advance led to the next stage of development in military flight simulators in 1982. The Coupled Airbourne Systems Simulator (VCASS) saw the birth of the modern Head Up Displays (HUDs), which coupled a helmet mounted display system to the external CG environment displayed on screens. This meant that flight and targeting information could be overlaid over the CG landscape, which would later allow different information to be made available dependant on where the pilot looked, e.g. looking at the wing would bring up information on how many warheads remain (Furness, 1987). In 1984 NASA arrived on the scene with the Virtual Visual Environment Display (VIVED), a stereoscopic HMD producing monochrome stereo paired displays, with tracking that allowed the user to look around the VE (Fisher, 1983). Later editions developed in the Virtual Interface Environment Workstation (VIEW) would incorporate VPLs power gloves, and sound and voice recognition to further increase immersion (Fisher et al., 1987).

VPL Research, a company started by Jaron Lannier and Tom Zimmerman, enhanced Thomas Defantis' Sayre glove, creating what would be commercialised as the Data glove. VPL have an important place in the history of VR as they were the first to offer commercial VR systems. These included hardware like the Data glove, the Eyephone, (Figure 4.9) and a full body suit, along with software for modelling, and programming dedicated to the creation of VEs. Their efforts culminated in the creation of a shared VE for two people (Blanchard et al., 1990; Lanier, 2010; Zimmerman et al., 1987). These devices were used by NASA among others, and mark the start of VR development outside of government funding streams. VPL were not the only company to benefit from VIEW and NASA contracts. In 1989 Fake Space Labs created the Binocular Omni-Orientated Monitor (BOOM). The Cathode Ray Tube (CRT) stereo pair displays were mounted on a mechanical boom arm, that translated both the position and the orientation of the device back to the VE. By looking through the displays a user could see into the VE, and by 1990 could use an adapted VPL data glove to manipulate Computational Fluid Dynamics (CFD) data around shuttle models (Figure 4.10) (Bryson, 1993).

Figure 4.9: NASA used the commercially available VPL EyePhone and data glove BBC, 2013

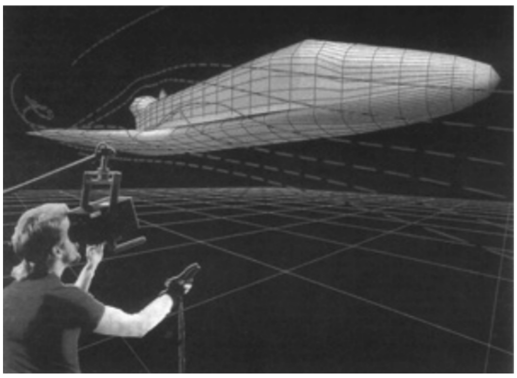


Figure 4.10: A mock up of Fake Space Labs BOOM device used in NASAs Virtual Wind Tunnel Bryson, 1993

HMDs still had a number of drawbacks in the 90s as they were heavy, low quality, low resolution, single user, high latency and expensive (LUDWIG, 2010; MAZURYK et al., 1996). To help address this the Computer Assisted Virtual Environment (CAVE) was created at the Electronic Visualisation Lab at the University of Illinois at Chicago. The CAVE uses a back lit projection system to display four or more surfaces around a user with sequential stereoscopic displays. LCD shutter glasses synched with the system and allowed for the correct image to be received by each eye (CRUZ-NEIRA, SANDIN, DEFANTI et al., 1992; DEERING, 1992). The advantage of being surrounded on 4 or more sides by 3D displays was that; more than one user could work in the VE at once; a wider field of view was possible; there are no cables to restrict freedom of movement; and the wearable devices were comparatively light weight compared to even current HMD equivalents. With multiple users now able to be within the same VE, groups could share and collaborate their work with ease. NPSNET and AVIARY pushed the boundaries even further by allowing users to connect from remote locations over the internet (MACEDONIA et al., 1995; D. N. SNOWDON et al., 1994). These can be seen as the fore-fathers of all Massively Multi-player Online games (MMOs).

Graphic engine development in the 90s underwent vast improvements to compensate for the limitations of hardware that still lacked processing power. In 1996, a state of the art Reality Engine Silicon Graphics Card card could process 1 million textured polygons at a time (MAZURYK et al., 1996). Modern graphics cards and engines no longer use polygon or triangle counts in their marketing since they are no longer considered a bottleneck (NVIDIA, 2014). For reference, in 2012 a commercial gaming graphics card could render over 12 million polygons (2 triangle) at 40 frames per second (CHAUHAN, 2012). The advances in graphics engines in the 1990s led to techniques for image skewing and adaptation, such as image deflection (MACIEL et al., 1995; SO et al., 1992), z-buffer warping (S. E. CHEN, 1995), address recalculation pipeline (REGAN et al., 1994) and frameless rendering (DAYAL et al., 2005), which were all developed to reduce processor load. These techniques would prove vital for future development of HMDs, and game engines in the years to come.

As the graphical ability of systems increased, universities started to produce VR software toolkits similar to VPL's Body Electric, ISAAC & Swivel (LANIER, 2010). These new bespoke tools, such as Minimum Reality (SHAW and GREEN, 1993; SHAW, GREEN et al., 1993), NPSNET (MACEDONIA, 1995; ZYDA et al., 1992), AVIARY (D. SNOWDON et al., 1993), DIVE (CARLSSON et al., 1993), allowed a broader skill set of staff and students to use VR in research for a wider set of reasons. Coupled with the reduction in price of the associated technologies, this allowed university and industrial research departments to purchase equipment to develop VR. The 90s saw the use of VR diversify from what was predominantly military research into an additional five areas. These were architecture, the sciences, medicine, manufacturing, and design.

An example of this diversification was at the University of North Carolinas (UNCs), where VR was being used for Architectural walk-throughs. This led to further development of HMDs, optical trackers and the first pixel plane graphics engine (BROOKS JR, 1987). The work at UNC branched out into other universities and institutions around world including:

- Graphics, visualisation and usability of VR at the University of Georgia (ALLISON et al., 1997; BOWMAN et al., 2001; HODGES et al., 1995) centring around human interaction with the VE, a Virtual Gorilla for education and VR for treating Phobias.

- The University of Houston Virtual Environment Technology center collaborated with NASA in training, scientific data visualisation and education. The lab was also involved in the US program of looking at the uses of VR in military training (VETL, 1997).
- Technology was developed at the University of Illinois at Chicago including, as previously mentioned, the CAVE & Immersadesk (CRUZ-NEIRA, SANDIN and DEFANTI, 1993; CZERNUSZENKO et al., 1997).
- The Fraunhofer institute in Germany researched medical and lighting calculations in VR (MÜLLER et al., 1995; SCHÖFFEL, 1997)
- The Iowa Center for Emerging Manufacturing Technology (ICEMT) at IOWA university concentrating on Engineering applications, including dynamic statistical data exploration (ARMS et al., 1999), immersive architectural modelling (CHAN et al., 1999), a driving simulator (GRUENING et al., 1998), and virtual prototyping (NIESEN et al., 1999)

From research groups like these the applications of VR have grown, into what was to become the predominant areas of research through the 2000s. One of the most successful areas of the application of VR can be better described as what Thomas Caudell called Augmented Reality (AR). Caudell first coined the phrase Augmented Reality in 1992, while discussing the synthesis of the virtual world and the actual (CAUDELL et al., 1992). Its use today in medicine is one of the unmitigated success stories of the technology.

Although the graphics engines of the time were not powerful enough to give any sense of verisimilitude, it was found that enough could be displayed to allow doctors to operate using tele-operators (BALLANTYNE, 2002; MARESCAUX, SMITH et al., 2001). Laprascopic surgery assisted by VR or AR systems has proven to be popular over the years due to its minimally invasive nature. This significantly reduced the chances of infection, the pain experienced by the patient, shortened the time patients needed to spend in hospital and increased the postoperative immune functions (ALLENDORF et al., 1997; FUCHS, 2002; V. B. KIM et al., 2002). Using computer-operated arms allowed for the smoothing of the surgeon's movements but compromised on the full degrees of freedom they would normally enjoy. The use of a VE permits a surgeon to be trained using these tele-operated robots (such as the DaVinci system) on models that firstly, will not die and secondly, can be calibrated in any way required. VR has also been used in psychology to treat phobias and PTSD (DIFEDE et al., 2006; EMMELKAMP et al., 2001; GERARDI et al., 2008; ROTHBAUM et al., 2002). The latter has proved particularly affective for treating military personnel. The wide spread use of MR in medicine especially, shows that the technology has matured. If it is deemed safe enough to use on people, and can be certified safe to do so (FDA, 2018; FDA, 2017), then it may be of a Technology Readiness Level (TRL) at which it could be certified for use in nuclear manufacturing. Furthermore the systems used in medicine can increase the levels of control and fidelity that can be produced by users. In an industry that has fine tolerances, this is relevant.

In architecture, the use of VR diversified into two main user groups, that of the internal users, predominantly designers, and of the end user or customer (WHYTE, 2003). For internal groups, the ability to manipulate the data was more important than the acuity of the simulation. For the customer or clients, acuity and a high enough frame rate for real time movement were paramount. This is because one of the proven causes of MR sim sickness is jerky movement caused by inconsistent and low frame rates (KOLASINSKI, 1995). The real time solutions for high frame rates can be found in game engines, where high frame rates are also necessary (SHIRATUDDIN et al., 2002). The technology has been transferred from being used to look at what could be built, to what has been built (GAITATZES et al., 2001). Recent advances in Building Information Modelling (BIM) and Computer Aided Design (CAD) have given the architecture profession new tools for creating buildings. BIM can offer designers an interactive design space, so that buildings can be designed from the ground up in a VE (LEVY, 2012). It is hoped that with new generations of CAD skilled engineers and designers coming through, 3D and VR would become more prevalent in BIM (MOBACH, 2008). The use of BIM for manufacture facilities is further discussed in Section 4.2.

Within the sciences, large scale data representation in MR is used to display complex arrays and inter-relationships of data; from testing how the electronic signals work in the human brain (BAYLISS et al., 2000; KU et al., 2003), to molecular constructs and protein docking (A. ANDERSON et al., 1999; DAS et al., 1993; HAMDI et al., 2008). As the amount of data generated within scientific study increases, effective methods for displaying it in a comprehensible format becomes increasingly important. A VE offers somewhere to gather interrelated data sets, in a fashion that converges with reality to aid understanding. Programs like PyMol, the virtual wind tunnel, and planet surface scans all convert large and difficult to comprehend data sets into a VE where comprehension is more natural (BRYSON and LEVIT, 1992; DELANO, 2002; HITCHNER, 1992). The wide variation of input and output methods in VR, coupled with the ability of additional non-real guides or distortions of reality, can aid understanding (BENFORD et al., 1995; MAES, 1995) and enhance human perception and information retrieval (CAUDELL et al., 1992; HAASE, 1996; MAPLES et al., 1995).

In the 90s the cost of equipment & the limitations of bespoke software excluded the user from MR development. It was not until the late 2000s that cheaper electronics and developments in the mobile industry made MR development possible for non-professionals. It was in the early 2000s that enthusiasts started experimenting in their own homes and the internet was widely used to share and disseminate experiences and information with others (MTBS3D, 2014). This led directly to the start of what has been heralded as the VR Renaissance (LUDWIG, 2010).

Table 4.1 shows the dramatic reduction in price and increase in capabilities of VR technology over the last twenty years. The technological advance has not only been limited to hardware. Software to display VEs has made significant progress from being developed in-house on predominantly unix based systems, to being adapted from existing high-end gaming engines. The latter has all the advantages of the support of a multi-million dollar game industry which is constantly improving the software. One of the significant advances in this field was the adaptation of the Unreal Tournament 2004 engine for the CAVE environment (JACOBSON, 2005). The use of this engine allowed game developers to create content for Immersive Virtual Reality (IVR) without having to significantly adapt their skill sets. Although the development of CaveUT has ceased, the PublicVR group continue to work on VE for education using the Unity Engine & MiddleVR (KUNTZ, 2014;

Table 4.1: A comparison of HMDs from the 1990/2000s with examples from 2014

Model	Year	Price US$	Panels	Resolution per panel	Colour depth	FoV horiz.
Flight Helmet	1991	$6000	LCD	240x120		
Tornado jet simulator	1995	N/A	4 panel CRT	1280x1024	true col.	127°
nVisor SX	2003	$23,900	FLCOS	1280x1024		60°
I-glasses SVGA Pro 3D	2005	$1,199	2 panel LCoS	800x600	24bit	26°
nVisor ST (AR enabled)	2005	$34,900	2 panel LCoS	1280x1024	24bit	50°
Carl Zeiss Cinemizer	2014	$749	0.39" OLED	870x500	true col.	30°
Oculus Rift DK2	2014	$350	7" OLED	1920x1080	true col.	100°
Silicon Micro Display ST1080	2014	$799	0.74" LCoS	1920x1080	true col.	39°
Sony HMZ-T2	2014	$894	0.7" OLED	1280x720	true col.	45°
Oculus Rift CV1	2016	$399	OLED	2160x1200	true col.	110°
HTC VIVE	2016	$599	OLED	2160x1200	true col.	105°
OSVR HDK2	2015	$399.99	OLED	2160 x 1200	true col.	100°

TECHNOLOGIES, 2014). MiddleVR is an middleware program that converts the Unity game engine for use in HMDs, Domes, and CAVE systems. A more recent iteration of has been the conversion of the more powerful UDK engine, which is built on the Unreal Engine 3 (LUGRIN et al., 2012). This offers higher frame rates, and a free Software Development Kit (SDK). The use of game engines is now commonplace within serious gaming academic groups, and as VR technology progresses, its use in education is becoming more prevalent (JACOBSON and GILLAM, 2012; NAKEVSKA et al., 2011). This has led to further work in reducing the cost of VR, with the use of the CryEngine2 (JUAREZ et al., 2010) and Quake III Arena (DOMAGOJ BARIČEVIĆ, 2007), both being ported for use in low cost DIY CAVE systems.

Game engines offer the benefit of off-the-shelf technology with advanced computer graphics and in-built physics engines. More recently, game engines have been shipped with their own development environments and online tutorials, making the creation of VEs more accessible. It can be expected that as the price of hardware falls further, a skills transfer from the much larger game industry into MR development could take place.

The use of game engines is a mixed blessing, as it comes with the drawbacks associated with using software developed for a specific use. Although large companies are backing VR they are predominantly looking at its application in the games industry, the needs of which are different from that of the nuclear manufacturing community. Games engines and VR engines share the requirement of high frame rates, convincing visual affects and physics. This is not a new thing for computer graphics, as Ivan Sutherland was involved in the first stages of freeing up CPU cycles by using occlusion to reduce the amount of processing power an engine needed to render. To achieve these in a game, the Level of Detail (LoD) is reduced and texture mapping is used to replace or spoof geometric data. In fact, it is often said that the sign of a good games designer is someone who knows how to do more with less, that is, to convince the user they are seeing something in a game, when they are not. This was achieved in early games by using 2D sprites to represent 3D objects. Computational power has increased since sprites were last commonly used as 3D objects, however the demand for complexity, interactivity, and high end graphics has also increased. This has led to further developments of graphical methods to reduce the load on CPUs and GPUs in a system. One such technique is to use bump maps to show a flat surface as one with relief. In this way, the object that has the bump map looks as if it has a much higher level of geometric detail than it actually has (Figure 4.11). The computer thus has to deal with less geometrical data, but does have an additional level of lighting calculation that must be done.

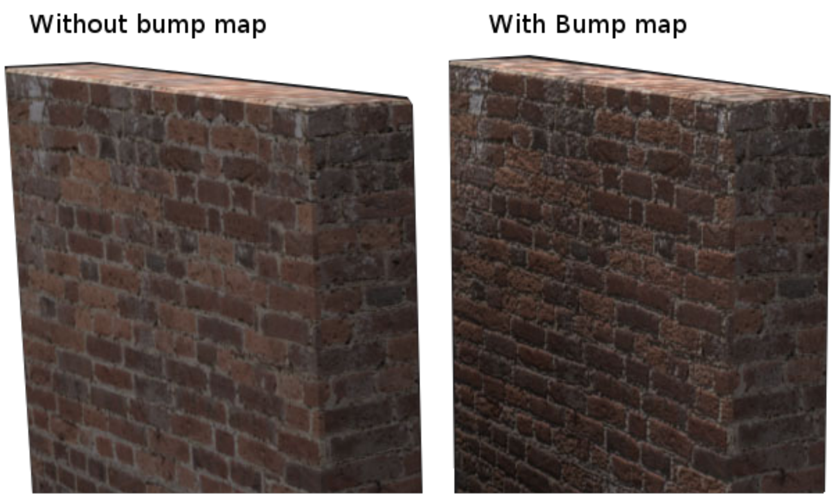


Figure 4.11: Use of bump mapping on a rendering of a brick wall (MAX et al., 1994)

Techniques for speeding up the processing of time, and allowing higher frame rates, are beneficial for MR in gaming and simulations. Techniques like bump mapping or normal maps that use light calculations will not work when two camera positions are required, making them unsuitable for near field stereoscopy (FLOCK, 1965) (HOWARD et al., 1995). The ability to spoof geometric detail has led to engines being developed that are more focussed on lighting calculations than geometric meshes. This means that objects imported into game engines are normally stripped of as much geometric data as possible, which is done by reducing the vertex count in the meshes. The reduction in geometric detail can run contrary to what could be required in simulations. This would be the case if MR was being used to check the fit between two parts for assembly, or running physics engines using the objects' centre of gravity. Physics engines are resource intensive as the calculations need to be done on all affected objects within the VE. The number of affected objects can only be reduced by specifying pertinent parts that require interaction. In game engines, this is taken even further, where the use of the physics engine is reduced as far as possible, so that water and lighting affects are calculated using optimised algorithms that do not conform to physics models. Most of the lighting in a game engine is baked into the textures rather than calculated in real time. As with the geometric data, this means that any simulation that would require lighting, would need to specify that real time lighting was calculated in the relevant regions of the VE. There are many other methods that game engines, and game environment developers use to reduce the computational load required to run the VE. These issues illustrate the challenge that will need to be faced if MR authors are going to use game engines to create their VEs. As promising as having a new developing group working on MR is, it must be understood that the goals of the games industry are different from that of the manufacturing industry.It is therefore important to be as aware of the engines' limitations as the game developers, and work around them when creating VEs. These challenges are often applicable to other MR engines, however as these engines are created specifically for MR in industry they can be better adapted to industrial requirements.

In MR development it is often the military who have been able to afford the most capable systems, and for IVR experiences for training this is certainly true. The impressive VIRTSIM (VIRTSIM, 2014a; VIRTSIM, 2014b) is being used by both the US military and Police forces. The systems work using full body motion tracking and HMDs, allowing for combat scenarios to be tried, tested and evaluated both in real time and post exercise. Another simulator, Virtual Battle Space 3, offers desktop to fully immersive scenarios, and is being used by globally (BISIM, 2014). MRs are currently being used for training, command, and control of forces around the world (KNUTZON et al., 2004), UAV pilots (WALTER et al., 2004), and medics (VOZENILEK et al., 2004).

4.1.5 Discussion

> "The biggest thing that's happened is that the industrial use of VR — as opposed to consumer and entertainment use — matured, and has become ordinary enough to be boring. But it hasn't happened in this sort of big, unified way; it's a bunch of little pockets that are each very specialized." (LANIER, 2014)

Early development work in both CG and VR systems was both motivated and funded by the military: the funding from the Department of Defence (DoD), US Army and Air Force leading to the creation of academic research groups whose expertise and knowledge were to develop in the same areas. Novel applications outside of military uses were found for technology transfer, and alternative funding routes for research groups within universities. Scientists like Fisher, Lanier, Zimmerman and many more started small spin-out companies which then commercialised the VR technology. Unfortunately companies like VPL and LEEP did not diversify their markets, and so closed in the late 90s as the technology failed to deliver on the hype and promises. The fast pace of advancement in mobile technology since the 2000s has led to an influx of affordable technology capable of transfer to VR applications. In turn, this has led to the growth of a grass roots movement of enthusiasts developing their own VR headsets (MTBS3D, 2014). One company to emerge from the grass roots community was Palmer Luckeys' Oculus VR, which was acquired by Facebook for \$2 Billion (LUCKEY, 2014), and is now a major software and hardware developer for HMDs, alongside the resurgence of in house development from Samsung, HTC, Valve and Sony. The involvement of larger companies has led to greater development of the VR toolset. Bespoke VR tools may no longer be the norm as game engine developers collaborate with manufacturers to create functional VR engines complete with editing suits. The reduction of cost and bespoke software solutions coupled with the rise of affordable or free generic software for the production of MR has served to make MR accessible to a greater number of creative people.

Today the majority of work and development in MR is still done by four main parties:

- The military, with the largest budget, and the most advanced immersive systems (EISLER, CHATTERJEE et al., 1996),
- University research departments working on fundamental research, often with military funding, and the transfer of technology to other existing areas of research (CRUZ-NEIRA, 1998; SCHUEMIE et al., 2001),
- VR enthusiasts developing on a small scale at home and nominally working towards gaming or metaverse outcomes,
- Larger companies that have either developed out of enthusiast groups or existing electronic companies.

With a reduction in the price of the required technologies, and as the software involved becomes less bespoke, and so more multifunctional, previous undeveloped areas will see MR applications created for them. One of these areas has been manufacturing, but the application of MR in this field has been limited, as the evidence presented in Section 4.2 will show.

The research available from the military is often restricted, with university research departments working within defined areas set by a funding criteria. VR enthusiasts doing development on their own are rarely concerned with producing papers but do publish on websites, whilst large companies prefer to keep their work behind closed doors and rarely publish articles. This combines to create an arena where today publicly available and reliably peer-reviewed information is limited.

The use of MR in Manufacturing has only reaped benefits for MR and manufacturing in the last two decades. Despite the advent of DAC in 1964 (JACKS, 1964), and the development of CAD system since then, it was not until the boom of the early 90s that MR was to be applied to the challenges in manufacturing.

4.2 Virtual Reality in manufacturing

The use of VR in manufacturing is being developed in areas where visualisation of data sets and information to aid understanding is important. It is now commonly used in the design cycles or iterations. Significant research has also been done, not just in product design, but also on factory layout, cell functionality, and material resource planning (H. ELMARAGHY et al., 2012). There are currently three main objectives when using MR in manufacturing: the evaluation of designs, their assembly processes and sequences, and the creation of training simulators for operators.

4.2.1 Design

Rapid prototyping of designs has been found to be complimentary to the design process (GIBSON et al., 1993). The ability to create mockups of product designs before manufacture does not conflict with a designer's process, but instead aids the designers in understanding issues and identifying new potential challenges with the product.

VR offers a number of advantages over design prototyping with physical mockups. Physical mockups lead to increased costs in the product design phase, and consequent delays in bringing the product to market (MA et al., 2009; SHAHMANESH, 1998). The removal of physical mockups reduces the design cycle time, and improves adaptation to changes in the market (BURDEA et al., 2003; G. G. WANG, 2002). This allows for greater flexibility within the company. By removing the manufacturing costs and time required for creating physical mockups, the company can not only change existing products to meet new customer demands, but also design new products in a shorter time. It allows removal of design and development risks early on in the design cycle, reduces the cycle time and therefore the cost(KALAWSKY, 1993). VR allows anyone to interact and review a product before manufacture or physical prototyping takes place. Customers who are not always able to understand complex engineering diagrams or CAD data, will be able to view the product in a suitable environment before production begins. Thus customer requested changes can be implemented before production, which can allow for greater customer orientation in the business.

Using VR in design requires the conversion of existing CAD into models suitable for VR engines and display systems. This can be a time-consuming operation requiring skilled operators and 3D modelling tools. It is estimated that 70% of the time creating a VR application is spent on preparing the CAD data, with the remaining 30% spent authoring the VE. The converted polygon models often only contain the geometrical data that are used for spatial and collision calculations. The topology, physics, material, and semantic data is not translated across, and need to be attributed separately as meta data for each part. This can be done manually within the VE by the creator (NEUGEBAUER et al., 2011) or by connecting the VE to a Product Data Management System (PDMS) (BARBIERI et al., 2008). Barbieri et al. created the Virtual Design Data Preparation system (VDDP) that allows the user to import data from the PDMS while in the VE, and which allows for model optimisation within the VE (GUENDELMAN et al., 2003). For the Digitally Assisted Assembly Cell (DAAC) any cloud conversion techniques or conversion programs that require the information to be uploaded to external servers is going to be an intrinsic security risk, and as such was not considered.

Figure 4.12 shows a standard setup of data integration with a VE. An abstraction layer is set-up, in this case a Product Data Management (PDM) system. The bottom layer represents everything that would be the geo-spatial and meta data within the VE, it is imported and stored within the PDM as data sets and tables that can be polled and updated as and when required. The cyclic information flow between the PDM and its subsystems is important as it represents the constantly changing nature of any system. The CAD system can also update the models stored within the PDM as required. The important aspect of this diagram is that the data flow is bi-directional with all systems except for the VR system. This is partly because of the preparation of the data sets for use within the VE, as data need to be formatted and stored in both the correct syntax, but also format for import into whichever VE engine has been chosen. Within the S-Frame (Chapter 6) project a JSON data format was implemented, and while the data flow was uni-directional, the information was captured within by sensor sets and pushed to an AWS server (equivalent to PDM in this case)

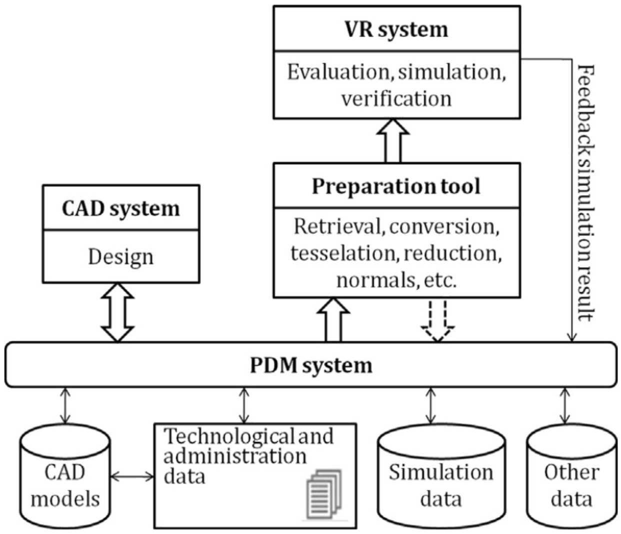


Figure 4.12: Data flow between CAD and VR systems (LEU et al., 2013)

where the VE polled it at regular intervals. Simulation update results can be fed back into the PDM as required.

A variety of other methods have been created to import non geometric data, using Autodesk Inventor (Q.-H. WANG et al., 2010), or retrieving data from a custom database (BOWLAND et al., 2003). Another approach is to create the data within the VE itself, eliminating the need to import it from external processes. This has been attempted for use in architecture and more recently in the design of modular housing.

Architectural design in IVR has been in use since UNC created the first walk-through project (BROOKS JR, 1987). Until recently architectural design thinking has been narrowed by the fact that the designer is restricted within their process by the inability to experience their designs in full scale (SCHÖN, 1983). Using a Virtual Architectural Design Tool (VADET) within a CAVE system, it is now possible to design architecture and experience it at the same time (CHAN et al., 1999). A major drawback of this system was found to be that the designer was unable to observe the full scene at once, and a lack of experience of the user working in a full scale environment. However more integrated solutions such as AutoDesk Stingray have supplied a new generation of designers with access to their models in VR (AUTODESK, 2017b).

Recent advances in MR and AR have led to an investment in research by both the manufacturing and construction industries. The possibility of providing workers with data overlays onsite has capture the imagination of funders at innovate UK (AMRC, 2017a), Destaco (AMRC, 2017b), and research streams from the Universities of Sheffield and Strathclyde (CLEVER-MACHINERY, 2017) (REALM, 2017) (AREA, 2017).

Design evaluation and modification of systems in VEs is not limited to architecture. Mechanical systems used in civil engineering have been successfully linked to VEs for evaluation and modification. A bi-directional system using both multi-body dynamics and particle simulation to assess mechanical designs was created by Antonya et al. (ANTONYA et al., 2007). As with other similar systems, the current limitations are set by the processing power required to run the simulations in parallel with the VE (EBERHARD et al., 2006; KARKEE et al., 2011). When integrating disparate data sets into a VE, the fidelity of the images is not enough: a high accuracy in the simulation itself is required. An example of this is in collision detection, where the simulation needs to maintain the same frame rate as the VE, or the environments will lose synchronisation (VINCE, 2004). This can be solved by frame rate synchronisation, which is used in modern engines with inbuilt collision detection. Synchronisation techniques would need to be created for non-native collision calculations.

A more successful combination of data sets has been that of remote collaborative environments pioneered by systems like NPSNET and AVIARY. Designers can now work across distributed systems (LEHNER et al., 1997), which offers a variety of challenges, some unique to VR, and others based on the nature of collaborative design in real time. The latter revolves around version control, and ensuring that each designer is working on a separate area of the product, and as the system updates, work is not over-written. Unique to VR are the new methods required to work within an VE: data gloves, wands and space mice are used instead of more standard computer accessories. This requires a level of re-training or adaptation from the users, which is highlighted in both Co-CAD (GISI et al., 1994) and Syco CAD (NAM et al., 2001). The rise of 3D art programs such as Tilt Brush from Google, demonstrates that this re-training is not insurmountable (GOOGLE, 2018a)

The ability to design, modify and collaborate in a VE has recently been achieved by Wang et al. (Q.-H. WANG et al., 2010), by coupling Autodesk Inventor with COM using the Application Programming Interface (API). The link with COM allows bi-directional communication between the VE and the Inventor, which allows real time manipulation and design changes within the VE. STEP files are used, as they are an open format that supports the persistent naming mechanism, which allows for meta data on how the parts are linked to be transferred directly from CAD to the VE. This work is proof of the concept that the possibility now exists to use VEs to provide a quick design change review and evaluation of products.

VR is not limited to the designers' interaction alone, as Mercedes and other auto manufacturers use customers as part of the design process, by allowing them to sit in a virtual vehicle and give their feedback to the designers (TAMURA et al., 2001). The ability to have customer feedback on a model before production begins, can have clear advantages.

The Fraunhofer Institute is one of the world leaders in Virtual design and manufacture and is working with a wide variety of companies and institutes (FRAUNHOFER, 2014). As a group they have led the development of MR in design for the German auto mobile industry for the last decade. Most recently the Arena 2036 is the creation of an *"Active Research Environment for the Next Generation of Automobiles"*.

4.2.2 Manufacture

Manufacturing applications of VR include process simulation, factory layout, assembly methods, prototyping and part flow simulation (MAITEH et al., 1999). Within manufacturing, a wide variety of simulation software is used to help create efficient factories and production facilities (Section 3.4.2. The design and layout of facilities has been created using architectural CAD, and so similar examples of the use of VR in design can also be applied to manufacturing facilities. Factory design and layout is important in order to maintain operational safety and the efficient use of resources. The ability to assess these aspects before the construction of a factory emulates the benefits found in product design (SOUZA et al., 2006) (BOWYER et al., 1996). Combining factory and product models together with a Time Integrated Management (TIM) system (ONOSATO et al., 1993) allows for production cycles and schedules to be trialled before implementation. Reconfigurable factories can be assembled, investigated, tested and evaluated before anything is physically built (H. ELMARAGHY et al., 2012). Existing factory processes can be optimised and evaluated within the VE as part of the production planning process and to ensure safe implementation. Work performed with Rolls-Royce (RR) on different factory layouts for the manufacture of nuclear pressure vessels identified a number of useful outcomes from the use of VE. These included identifying if there is enough space for all the planned components, product flow, transportation techniques, layout and product flow, manufacturing processes, complex lifting and many more.

These techniques can also be used on existing factories which predate the evolution of CAD models. To achieve this the 3D information of the factory, machines and products need to be obtained either from paper drawings which are then converted into 3D models using a CAD package, or by a data acquisition technique. Geo-spatial data acquisition can be separated into contact and non-contact methods. Contact methods include CMM, and measurement by robot; these use the spatial coordinates gained by translation of points of contact negated from the known 3D volume of the device. Non-contact methods involve optical and non optical methods, both of which rely on signal interference and/or correspondence to combine to create the mathematical constructs of the 3D geometry. Optical methods can be divided into 2D image based, and range-based techniques. 2D image based techniques convert single or multiple 2D images into 3D geometry. Example methods are shape from shading (HORN et al., 1989), specularity (HEALEY et al., 1988), texture orientation (KENDER, 1979), contour mapping (ULUPINAR et al., 1995), 2D edge gradients (WINKELBACH et al., 2001), shading and generic geometry (YOSHIKI et al., 2006) and multiscopy from 2D image sequences (DICK et al., 2004). Range-based techniques rely on live signal interference, e.g. measuring the disturbance in a defined light pattern (FECHTELER et al., 2007), which is a structured light scanning process (GUPTA et al., 2011). One of the issues with the latter method is pollution of signal from global illumination, which can be significantly reduced by using structured and monochromatic light (GUPTA et al., 2011). Lasers can also be used effectively in time of flight calculations to create large point cloud data sets (R. ZHAI et al., 2011).

A drawback of all data acquisition methods is hidden geometry. Any area that is hidden from the scanning device will not be captured and, in consequence, convex detail can be easily lost. To adjust for this, multiple scans need to be taken from different angles, but this can greatly increase the time required for the process. A hybrid or combination of both image-based and range-based acquisition can be used to create a method that has high geometric accuracy, portability, full automation, photo-realism, low cost, flexibility, and efficiency (BALTSAVIAS, 1999; BOEHELER, 2005; REMONDINO et al., 2005).

Geometric data sets, acquired from CAD or scanning techniques imported into a VE, can be combined with other data sets from a production facility. In 1994 Boeing started using an inventory and Material Resource Planning (MRP) integration tool within a VE, to help assess the inventory performance and release systems (H. A. SCOTT, 1994). This can also be done in real time, as factory data from the production facility is combined with a virtual counterpart. With the introduction of cameras and AR systems, a bi-directional system can allow more efficient control of a facility (HIBINO et al., 2006).

4.2.2.1 Discrete Event Simulation in Virtual Environments

Research has also focussed on Production Planning & Control (PP&C) by taking Discrete Event Simulation (DES) data and applying it to a factory setting within a VE. The advantage of using a VE over traditional 2D or 3D displays is how natural the error spotting process is, and how this reduces the task time. Tests carried out in 2005 found that users found over twice as many existing ‘difficult’ errors in a simulation when it was viewed in VR compared to 2D (AKPAN et al., 2005). This work was an extension of earlier work by Kelsick et al on the challenges of DES in VR for a manufacturing cell. Their work focussed on using exported SLAM II DES data (O’REILLY, 1995) to show time dependant productions sequences. The work identified a familiar issue, which was the lack of processing power to handle complex virtual scenes. To allow for this, methods for reducing the polygon counts of geometric objects were used and collision detection was not present in the first iteration (J. J. KELSICK et al., 1998). An updated report on the work was provided in 2003, with a geometrically more complicated model and collision detection (J. KELSICK et al., 2003). A major advantage of demonstrating the work flow through DES in VR is that non-DES-experts can understand and contribute to solving problems in factory design and layout and production scheduling. Current DES platforms such as Technomatix (SIEMENS, 2014b), and Delmia QUEST (DELMIA, 2014), support 3D but not VE, but they do produce data tables which can be exported as American Standard Code for Information Interchange (ASCII) which can be read into VR simulators. Dangelmaier et al. used funding from the DoD to do exactly this. The project used the ASCII data and Wolverine software Proof animation package (HENRIKSEN, 2014a) to show the effects of variation in High Level Architecture (DANGELMAIER et al., 2005). This study proposed an extension of work in which the data could be used as an overlay on the real world. One collaboration that has successfully tied DES with VR is Lanners’ WITNESS which allows the end-user to watch their DES simulation take place within their virtual factory (LANNER, 2015). The partnership adds the Virtalis Visionary Render software to the backend of Lanners DES software WITNESS, which allows the import of complex CAD drawings, collision detection and interaction within Visionary Renders VE.

The use of static simulation using DES data sets in VEs can be used in the evaluation of the layout and ergonomics of the factory. Dynamic simulations from the same data set can be used to assess the process performance of the production system, the feasibility of the production plan and to help train operators in safety (W. ZHAI et al., 2009). The combination of data sets need not stop there: Zhai et al. proposed a fully integrated Virtual Factory Data Management System (VFDMS) in which the MRP and Enterprise Resource Planning (ERP) data sets are also included within the simulation. This would combine an unprecedented amount of data into one environment. The focus of the work was on how the systems would integrate rather than how the system would aid comprehension of the end user. The test case is a manual assembly line, where the level of detail and variation in data is limited, making it easier to parse and interpret. Another study in the field used the representation of DES in a VE to test and evaluate a die casting factory in Istanbul (BAL et al., 2009). The project was designed to test the idea of a Holonic Manufacturing System, and whether it could help companies to become more agile by testing adaptations before implementing them in the real world. In this respect, the DES model tested the flexibility of machining centres and staff before the market changes. It allowed the user to quickly conduct controlled and repeatable experiments or manufacturing variations in existing or planned facilities. Bal et al. deemed VEs to have the following advantages:

- Visual animation where certain patterns (e.g. inventory build-up, blockages in flow) would be quickly seen;
- The ability to implement identical production lines providing side-by-side comparative visual feedback on the operational differences;
- VR systems allowed the users to interact intuitively with the VE and its objects as if they were real, by immersing them in a highly realistic 3-D environment (BAL et al., 2009)

The ability to run experiments quickly was hampered by the fact that the DES software was not linked to the VE. Instead, simulation data was stored in an ASCII file that is then interpreted by the VE. This meant that any revelations made by the use of VR would need to be fed back and re-simulated, which would delay the results and reduce the number of variations that can be simulated over the full design or test period. The issue of two computationally intensive simulations being run in parallel was discussed in Kibira et al. 2002 paper(KIBIRA et al., 2002).

A solution to overloading CPU processes by combining DES and geo-spatial simulation was proposed and tested by Strassburger et al. (STRASSBURGER et al., 2005), who analysed three different methods of computational coupling of geo-spatial simulation and DES (Table 4.2).

Table 4.2 compares the levels of interaction that have been accomplished in several different DES to VR projects. Strassburger clealy favoured a bi-directional link between the simulation software and that of the VR. One variation that is not mentioned is that the DES calculations could be fully integrated into the VR software. That way, cross-system optimisations could be found, although it can be assumed that a high level of complexity and hence a computationally heavy calculation would be involved in such a task.

The emphasis is on real time response between simulations, so that users can interact and view the recurrent changes in DES caused by actions taken in the VE. A buffering approach is found to be most effective, as it created the lowest response time in ‘temporal parallelism’ and so it could be

Table 4.2: Feature characteristics with examples of Coupling VR & DES adapted from (STRASSBURGER et al., 2005)

Feature Characteristics	Examples
Post-run /unidirectional /monolithic /external	Visualization tool Proof Animation (HENRIKSEN, 2014a)
Post-run /unidirectional /distributed /external	Coupling of SLX simulations with visualizations in a cave (REHN et al., 2004)
Concurrent /unidirectional /monolithic /integrated	Visualizations in the tools TechnoMatix (SIEMENS, 2014b) and ARENA (ARENA, 2014)
Concurrent /unidirectional /monolithic /external	Visualization tool Concurrent Proof (HENRIKSEN, 2014a))
Concurrent /bidirectional /distributed /external	Coupling of eMPlant simulations with VR (BERGBAUER, 2002; FRANKE, 2004)
Concurrent /bidirectional /monolithic /integrated, external	Training simulators for pilot training

said to exhibit the highest level of concurrency. A buffering approach allowed for a fluid image generation within the VE which helped maintain a feeling of 'presence'. This was aided by the DES software choices of SLX (HENRIKSEN, 2014b), VRJuggler (CRUZ-NEIRA, 2013) and a custom VE, all of which were are less computationally intensive than many of their counterparts.

The study relied heavily on DES data, and so also suffered from a common issue of other simulations: the more the data had a low convergence with real world data, the less relevant the results were. One suggestion for checking the relevancy of data and results is integrating data sets from a real factory was proposed by Hibino et al. Using their idea of a Distributed Real Manufacturing Simulation Environment (DRMSE), combined with a Distributed Simulation Environment (DSE), the proposal was that the data used and created in factories such as machine use, cycle times, shift patterns, electricity usage, etc could be collected in real time into the DRMSE. This simulation was partitioned into separate groups, dependant on what the information refers to, and where it would be used. These abstraction layers were then interrelated with different layers of the DSE. This link allows real world factory data to be applied in the design of both product and factory. With a high level of coherence between the DSE and DRMSE then work proposed and tested in the DSE could be easily implemented in the DRMSE, and so transferred into the real factory much faster (HIBINO et al., 2006). Over time, as the factory data accumulated, the simulations could be adapted to increase accuracy until a level of confidence in the simulation allows faster adoption in the real factory.

One area where simulation is already playing a large part in manufacture is assembly, which can take up to 50% of production time and 20% of production cost (PAN, 2005). A great deal of research has been done in this area (RASHID et al., 2012; Z. WANG et al., 2009; ZHA et al., 1998). The research focuses on four main areas; assembly design (BOOTHROYD, 1994), assembly

Sequence planning (LAPERRIÈRE et al., 1994), system configuration generation (WEBBINK et al., 2005),and assembly line balancing (JUN et al., 2005).

4.2.2.2 Virtual Assembly Process Planning

In the last decade Virtual Assembly Process Planning (VAPP) has overtaken Computer Assisted Assembly Planning (CAAP) as the predominant area of research (Figure 4.13).

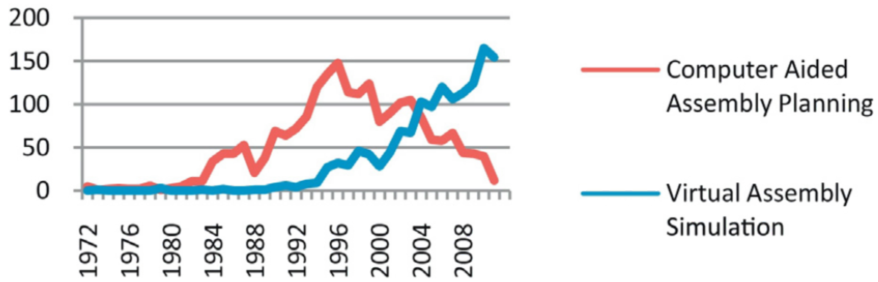


Figure 4.13: Publication count on CAAP and VAPP simulation (ENGINEERING-VILLAGE, 2012)

In 1994 the first VAPP system was created by Jayaram et al. the Virtual Assembly Design Environment (VADE) (S. JAYARAM, U. JAYARAM et al., 1999). The VADE system imported geo-spatial data directly from CAD, and required significant re-adjustment for acceptable relationships between objects to be created. Their work allowed for the testing of assembly processes within a VE, but it offered no method for optimisation.

VAPP is more intuitive for the user and can increase the efficiency of process planning. Computer Assisted Assembly Planning CAAP methods focus on how components are assembled, whereas VAPP focuses on the assembly process within a VE. How the planning process is modelled or organised has an influence on the data organisation and the planning efficiency. Although it may not be immediately obvious, the data organisation within the model directly affects how optimisation algorithms can work, and how effective or efficient they can be. One common method of organisation is a Hierarchical Assembly Task List (HATL). Using geometrical positioning of objects to separate tasks into sub assemblies, a hierarchy of tasks is automatically created. Assembly algorithms are then used to optimise them into sequences, which are recorded along with the associated times (JUN et al., 2005). Other techniques consist of creating semantic relations between different parts of the CAD geometry(ZHENG et al., 2006) or Degrees of Freedom (DoF) and constraints for compiling the assembly process, (R. D. YANG et al., 2007). Bowland et al. integrated existing CAAP with a Manufacturing Assembly Process Sequencer (MAPS) which allowed for existing hierarchy and program data to be used in a VE. The setup is of particular interest as it also recorded the path volume that the parts were required to move through (BOWLAND et al., 2003). One of the main advantages of a VAPP is the integration of the human into the environment, creating a method for recording known assembly sequences, and the paths that the parts are moved through by the operators to achieve this. The integration of ergonomic analysis and addition of human factors is discussed in (BULLINGER et al., 2000).

The user interaction, however, is important for other kinds of optimisation, and for recording the real task time. It can be considered that there are two differences between the real and simulation: Time & Geometry. Both of these suffer in VR simulation as the times are calculated by a machine with no accounting for human movements, and the geometrics are not properly accounted for in the algorithms, or in the handling procedures. VAPP helps to address this by the addition of the user-created assembly path and sequence planning (LIAO et al., 2004). In addition, VAPP addresses the spatial elements of assembly process planning (SETH et al., 2011). Assembly sequences can be based on the features, constraints and assembly semantics of the product (YAO et al., 2006; H. ZHU et al., 2010). For example, VADE (S. JAYARAM, UMA JAYARAM et al., 2007) uses a predominantly constraint-based model. Such models have improved the performance efficiency of the users, but the systems lack integrated optimisation schemes (YE et al., 1999). However semantic-based models provide a reduction in the translation process time of the metadata into the VE and the easier optimisation by algorithm as the recording of the user interaction becomes a less critical part of the process flow for creation. Instead, assembly relations and constraints between parts are defined from the CAD, and then genetic algorithms can be used to optimise the assembly sequences and routes before they are imported into a VE (SUNG et al., 2009).

VAPP can form an integral part of the Product Development Process (PDP), but a lot of time is still taken up by converting the models for the VE. Various methods for minimising this time are discussed in (BARBIERI et al., 2008), where MockupDVi and Virtools are used in conjunction with MATlab and NX3 models. They also expound on the practicality of using VEs to check if the end-user has enough room to wield and utilise the required tools for assembly by answering the question of *"Can I get a screw driver to the screw"* before the final design evaluation is complete. This is of particular importance when considering which design to manufacture.

4.2.3 In service

The predominant use of VR in service is for staff training. Considering its proven use in education, this is hardly surprising (KAUFMANN et al., 2000; OTA et al., 1995; PSOTKA, 1995) Virtual Reality Training Systems (VRTS) are made up of four components:

- A task planning module for setting up the VE and training tasks
- An instruction module that describes to the user what tasks to be undertaken and how to accomplish them
- The simulation module itself, within which the VE and the user will interact and complete designated tasks
- A performance evaluation module, which will analyse the times and motions made within the simulator (F. LIN et al., 2002)

The focus of the majority of literature on combining VR and training is on assembly planning.

The majority of VAPPs do not consider part path optimisation, and have no way of training the user which paths to take when assembling (Bobrow, 1988; J. R. Li et al., 2005; Tseng, 2006). One of the first examples of Virtual Assembly Training (VAT) was the Immersive Planning Virtual Assembly Planning and Training System (I-VAPTS) which took Jayaram's techniques in VADE and incorporated training simulations for the users (Yao et al., 2006).

4.2.3.1 Virtual assembly training

A significant gain in VAPP over CAAP was the use of path profiles for part movement. Christiand et al. show that this is equally effective when applied to training. The Haptic-path Sequence Guidance (HSG) is a haptic feedback method that reports to the user when they are deviating from an optimal path. The system shows a performance improvement in terms of accumulated assembly time of 28.33% and a reduction of travel distance of 15.05% when compared with unguided training. Sequence-guidance (SG) mode alone increased performance by 15.33% for assembly time and 11.36% for travel distance (Christiand et al., 2007).

The Virtual Training Studio (VTS) creates a master/apprentice scenario where the users are taught by the VE (Brough et al., 2007). In this the VTS plays the role of instructor by offering videos and demonstrations of assembly. This can reduce cost as it does not require instructor oversight, allowing the user to train independently. By logging the user's interactions within the VE and the time taken to complete tasks, the SCENARIOs simulator creates detailed logs for analysis by students and instructors. This can be used for verification of knowledge in assembly. The SCENARIOs system also allows the users to record problem areas for future work (Gomes de Sá et al., 1999), further assisting in identifying areas for product or training improvement. Another system for recording performance in a VE is the Haptic Assembly Manufacturing and Machining System (HAMMS). Similar to HSG, it uses haptics to record the users actions, but does not inform them of the optimal path. Ritchie et al. concluded that Virtual Assembly Training (VAT) is most effective when the controls are intuitive and are as close as possible to real world motions (Ritchie et al., 2008). In an improvement to haptic tracking, Chrysoulouris et al. created a system that tracks the user spatially, Virtual Reality Simulation Program (VRSP) (Chryssolouris et al., 2004). The combination of full body tracking and spatial tracking, allows for human factors to be assessed accurately within the VE. When VATs interact with other programs like Jacks (Siemens, 2014a), it can create a detailed assessment of ergonomics and movement within the work place. This can be used for a variety of purposes e.g. optimising movements in a time critical work environment to reduce injuries (Haggag et al., 2013). Lockheed Martin have been using full body tracking and a CAVE system to test assembly and loading methods for the Joint Strike Fighters (JSFs) among other projects in their Collaborative Human Immersive Lab (CHIL), (L. Martin, 2014)

4.2.4 The nuclear industry

Literature pertaining to MR use in the nuclear industry is sparce, which is not surprising given the sensitive nature of some parts of the industry, and the fact that many of the manufacturing techniques and challenges are the same in nuclear as other high value low volume industries. One area that is unique to the nuclear industry is that of nuclear radiation. Refuelling operations are both expensive and time consuming for site operators. The safety case has to be developed over months, as the worker doses are calculated and worker rotation programs put in place to conform to dosage limits. As all workers have personal dosimeters, and there are many more permanently fixed around the reactor, the data can be collected from all of these during a refuelling exercise and transferred into a VE. The resulting VR simulation is currently used by Iberinco, having been developed with the Nuclear Engineering Department of the Polytechnic University of Valencia. The CIPRES Project (Calculos Interactivos de Proteccion Radiologica en un Entorno de Simulacion, i.e. Interactive Calculations of Radiological Protection in a Simulation Environment) uses the VR simulation complete with radiographic maps to both plan future refuelling exercises and train operators and protection workers on how to minimise personal and collective dose (RÓDENAS et al., 2004). The environment is also used to test and evaluate variations on the procedures to keep the risk As Low As Reasonably Acceptable (ALARA).

Similar to the above study, Tecnatom South Africa created a training environment for operators working on in service inspection (TECNATOM, 2011). The training used the VIRTOOLS package to create the VE, and Sensable Technologies Phantom haptic interface device. The haptics were used as a recording device, to ensure that proper application of the Metallographic Replica technique was carried out. This second example is of interest because it shows the viability of scaling training programs in VEs, whether it be plant wide systems training, testing and evaluations, or system personnel applying Non-Destructive Testing (NDT) techniques. The benefits can be grouped as;

- training in environments often inaccessible or unavailable
- low cost in comparison with real world training
- high variability in situations, locations and events
- augmentation of display with information provided to aid training, testing and evaluation
- unlimited repeatability

These are of particular importance in the energy sector where overheads and the costs of mistakes are high.

Finally, work on combining VR with nuclear manufacture is either not in the public domain or non-existent. In 2012 a DES simulation in IVR was carried out as part of the EU Framework Seven Project COPERNICO, and with the collaboration of Rolls-Royce Civil Nuclear. The results of the simulation were used to help guide, Health Safety & Environment (HS&E) decisions, redesign of factory layout and processes, as well as contributing to the re-assessment of the business case (FREEMAN et al., 2012).

4.2.5 Discussion

Virtual Manufacturing (VM) is an information structure for real manufacturing systems, as it can integrate available information and tools into a single accessible location or VE (NOVAK-MARCINČIN et al., 2010). The use of VEs within manufacture is hampered by issues of data transference between different bespoke pieces of software, as no standard format currently exists. There are also unresolved issues with making sure enough metadata is attached or translated across from the real world into the VE. However, with careful selection of use case, and the analysis of the capabilities of existing technologies, then effective use of VR can be found. An example of this is the mapping of energy use in a facility, Neugebauer et al. demonstrated an effective bi-directional technique for highlighting areas of energy use as well as the times and amount of energy use. The use of VR allowed effective communication within all layers of the company, and could be used to reduce energy consumption in both existing and new-builds (NEUGEBAUER et al., 2011).

In two out of the three areas of the life cycle of a product discussed, MR use is highly specialised in comparison with other areas of industry. Academic research is either not meeting the needs of industry or it is unable to convince industry to invest in wider scopes for the use of MR. In fact, other than the industrial tests of Jayaram and Lodding (S. JAYARAM, UMA JAYARAM et al., 2007; LÖDDING et al., 2011) VAPPs and VATs are not being used in industry. However the review of literature has found that in the last decade academics have created applications that could be applied, and should be more formally assessed by industry. VAPP and VAT now has proven benefit for industry, as does the combination of IVR and DES. Below is a brief summation of the remaining challenges facing, and benefits that could be reaped by industry in the three predominant areas of a products life-cycle.

Challenges in MR for Design

- 70% of the time creating a VE application is spent on preparing the CAD, with only 30% spent on authoring the VE *
- Running or simulating systems parallel to the VE is resource intensive, so that both the simulation and the VE are limited by the available computational power *
- Collaborative design environments suffer from version control limits
- The benefits of real time collaborative design over nominal independent and version controlled design are still unclear

Benefits of using MR in Design

- The reduction of design cycle times allows for the removal of design and development risks early in the design cycle
- A reduction in the design cycle time and therefore reduced overall expenditure
- Increased Customer orientation, by presenting data in a more familiar and comprehensible format
- MR can be used to connect the VE to a PDMS
- Bi-directional linkages allow for designers to be aware of existing constraints when they are first raised in the PDMS

* This challenge is common in both design, manufacture and in service and is included in all for completeness

- CAD software has been successfully integrated into VEs since 2010 (Q.-H. WANG et al., 2010)
- The Design phase has also benefited from the use of human factors, via motion and body tracking

Outside its use in VAPP, the implementation of MR is largely unexplored. The benefits are currently based on the replacement of existing design methodologies such as removing the number of physical mock-ups created. Although it can be expected that this would be the case in the earlier stage of implementation, is it possible that MR could be used as a disruptive technology? For instance, could the design cycle start to encapsulate areas traditionally associated with later development? Could the use of MR in design for manufacture considerations use data from the shop floor generated by machines and personnel reports on previous product lines? Could challenges faced later in production, such as scheduling load balancing, be considered or addressed in the design phase? These are pertinent questions that remain unaddressed.

Challenges in MR for Manufacture

- Assembly can take up to 50% of production time and 20% of production cost (PAN, 2005)
- 70% of the time creating a VE application is spent on preparing the CAD, with only 30% spent on authoring the VE *
- Running or simulating systems parallel to the VE is resource intensive, so that both the simulation and the VE are limited by the available computational power *

Benefits of using MR in Manufacture

- Time integrated management combined with the factory layout and product models can be used to demonstrate time dependant process interactions
- Reconfigurable factories can be assembled, evaluated and reassembled many times within a VE before being built
- Existing factories processes can be both evaluated and optimised
- HS&E evaluations of factory and staff behaviour can be assessed
- Real time integration of data sets from the factory (ORiN) coupled to a Virtual Factory

Manufacture shares a number of challenges with both the design and in-service phases. However, assembly has been clearly identified as an area where significant advances could be made using MR and where most development has been done. The integration of real factory data and virtual factory data promises to provide a range of benefits for the manufacturing industry, but none of these benefits have yet been realised. This is in part due to the computational bottle-neck created by the integration of many large data sets, and the level of investment required to retrospectively implement the data capturing and interpretation systems required. For new-builds, the cost of the network and data analysis computers could be incorporated into planned budgets, and even

* This challenge is common in both design, manufacture and in service and is included in all for completeness

mitigated by cost savings made by MR elsewhere.

Challenges of using MR in-service

- Users may be required to don motion tracking equipment
- Users perform operations in a lab or class room environment
- 70% of the time creating a VE application is spent on preparing the CAD, with only 30% spent on authoring the VE *
- Running or simulating systems parallel to the VE is resource intensive, so that both the simulation and the VE are limited by the available computational power *
- Integration of VATs with real factory data is yet to be fully realised

Benefits of using MR in service

- Training is possible in environments often inaccessible or unavailable
- Low cost in comparison with real world training
- High variability in situations, locations and events is easily created in VEs
- Augmentation of display with additional information can be provided to aid training, testing and evaluation
- Unlimited repeatability, either with or without instructors
- Performance of user can be recorded and assessed, or used to improve current methods

The use of VATs has been explored and the benefits well defined, but the relevance of research and its benefits need to be proved in a real world environment, to evaluate what benefits industry could reap from it. Future expected challenges in such programs would be the reduction of training and production time down to a level where it was cheap enough to be sold commercially, and to introduce both SDKs and Application Programming Interfaces (APIs) which would allow end users to create their own content. At present, the cost of most industry standard motion tracking hardware and software is still prohibitively high.

The in-service use of MR is probably the most developed and tested area of the technologies within industry. Of particular note is that the Haptic-path Sequence Guidance (HSG) system shows a performance improvement in terms of accumulated assembly time of 28.33% and a reduction of travel distance of 15.05% when compared with unguided training. Sequence-guidance (SG) mode alone increased performance by 15.33% for assembly time and 11.36% for travel distance (CHRISTIAND et al., 2007).

Repeatedly the issue of translating the geometric and meta data into suitable formats for use in the VE has been presented in papers. Although there is no current method of addressing this, work has been done to try and create standardised formats for VR. A platform addressing the hardware

* This challenge is common in both design, manufacture and in service and is included in all for completeness

integration side has been created in VRJuggler (CRUZ-NEIRA, 2013). The software side is as yet underdeveloped but high integrity conversion of file types, and meta data could be on the horizon with programs like Polyglot and Brown Dog (ISDA, 2014; NCSA, 2014).

The amount of research being applied specifically to the use of VR in nuclear manufacturing or the industry as a whole can be said to be marginal at best. Without the substantial project undertaken by Iberinco the area would be almost devoid of this interesting and upcoming technology. The similarities that the nuclear manufacturing industry shares with other manufacturing sectors that have used MR could mean that it is a prime candidate for the transferral of technology and techniques. To achieve this some of the unique challenges, such as licensing, and enhanced regulatory control, will also need to be addressed. However as in the case of the Iberinco project this challenge creates unique opportunities to exploit the nature of MR.

4.3 Conclusion

> "We used to say seeing is believing. Now we have to say experiencing is believing" (YOSHIA, 2014)

Within industry the transformation of data into information, and then further into knowledge through understanding is as important now as it was when Russell Ackoff first presented his ideas. Where graphs or simulations are currently used to create information from data, MR can be used to progress information into knowledge by helping people understand the complex relationships between information sets. Tools like MR are only effective when appropriately applied; MR could be useful in training someone how to dig a hole, illuminating the HS&E risks involved, schedule delivery of necessary parts, assess how and when these parts may fit in the hole; but ultimately it would not help in the actual process of digging the hole. The future of MR will depend on suitable uses being found for it, so that it creates value within business. Common methods already covered for creating value have been: the reduction of cycle times within design, manufacture and in-service work phases, the verification of designs and processes within a facility, and the incorporation of different simulations into a VE. So, where can the use of MR add value into the nuclear manufacturing industry?
The current challenges that need to addressed in the nuclear industry are:

- decommissioning the oldest nuclear power stations and equipment
- providing technical assistance and safety verification for the life extension of the rest of the generation fleet
- design, manufacture and build for the new nuclear build Gen-3+ reactors
- design and licensing for Small Modular Reactors (SMRs)

Applications of MR identified from the literature that would fit, either through transfer from an existing use or by re-purposing existing technology:

The use of MR in design especially VAPP for training personnel in decommissioning techniques to minimise exposure to radiation. The work done on the refuelling process by Iberinco, is a prime example, ensuring that all staff on-site are aware of and prepared for each stage of decommissioning. Appropriate design could be used for the testing and evaluation of specialised decommissioning equipment. Devices could be checked for access, and staff trained on the device before the design cycle was finished, allowing rapid reworking and re-testing of the design before proto-types were created.

The above methods could also be applied to providing technical assistance, and safety verification for life extension. An additional area could be evaluation of the process of installing either replacement parts or modifications to existing parts. Using a combination of body tracking and process simulation, the refitting of parts of the reactor could be assessed from both process and human factors before it took place. This is not entirely dissimilar from what is done at Joint European Torus (JET) (W. KIM, 1994), where simulations of replacing parts within the toroid are done in VR before taking place in reality.

The scope for the new nuclear build program may have diminished but the application of the reviewed technologies remains possible and potentially beneficial. In Nuclear Manufacturing the prominent challenges are:

- High conformance to codes such as American Society of Mechanical Engineers (ASME)
- Lack of highly trained staff
- An ambition to achieve zero component defects
- Rapid process change
- Flexible facilities and staff
- Wide range of components
- Specialist equipment for handling, & manufacture
- Slow uptake of innovative techniques
- Documentation
- Long lead times

There is some scope for the data capture of legacy parts and constructions, the use of laser scanning and other technologies allows for the capture of the geometric data. Similar to the wing skin in the DAAC this data can then be imported into a VE. From their it can be analysed and combined with prospective new parts and components. In this way upgrade methods, and replacement techniques for existing nuclear reactors could be tried and tested within a VE before any work commences on site.

As previously stated the use of MR can be used to inform every one of these challenges. It is expected that over the next decade the use of MR in manufacturing will be as prevalent as CAD is today, partly because it is an extension of CAD itself, but also because it illustrates the next evolution of many of the current simulations used in manufacturing already. However it is not possible at present to create a complete list of all the future development areas that could benefit from using MR. Instead a couple of possible examples are suggested for each, using the literature review as guidance.

High levels of control could reduce costly processes designed to meet ASME codes
The use of MR in medicine has shown that it can be effective for use when using remote control; in fact, higher levels of control can be implemented through a MR system than without one (Ota et al., 1995). If it is safe and licensed to use on humans, then the technology could be safe to transfer across to the nuclear manufacturing industry, which also has a strict level of regulation. The higher levels of control and precision could help reduce the cost of meeting ASME codes surrounding safety and staff. Instances could include data overlays of what materials can, and can not be used in conjunction with each other, to avoid cross contamination. Even simple instructions of what Personal Protective Equipment (PPE) is required for which tasks, could reduce the constantly keep staff aware.

Highly trained staff
The use of VAPPs & VATs within the nuclear manufacturing industry could lead to significant advances in assembly processes, and for training staff. The ability to record every step taken within MR, which can then be played back and evaluated, can be useful in assessing staff performance. It could even be used to assess performance against existing recordings or the performance of other members of staff already skilled so that they act like a performance benchmark.

Zero component defects
Creating a culture or atmosphere for zero defects within industry requires a well informed workforce that fully understands the issues at stake. MR can be used to not only inform the work force, but also require participation within the simulation to aid comprehension. Simulating the assembly processes within VAPP would mean that staff can be trained to perform the processes correctly through VATs. Linking DESs can be used to insure that resources are always available to reduce the likelihood of cutting corners or the use of ad-hoc solutions.

Rapid process change
Process simulation and the use of ERP and MRP can ensure that facilities are prepared for process change, and VAT can train staff for the change.

Flexible facilities and staff
This helps create a flexible facility, and a workforce whose knowledge about processes and procedures can be refreshed and updated as and when required.

Wide range of components
The ability to import any component into the MR, and author any environment to interact with them, suffers from the previously discussed drawbacks of slow import methodology, and a lack of metadata imported with the components geometry, but training simulators can help staff adapt to the wide range of materials.

Specialist equipment for handling, & manufacture
As with the wide range of components, VATs & VAPP can be used to help staff adapt to handling

specialist equipment and complex manufacturing processes. It could help to multi-skill existing staff to increase the flexibility of the workforce.

Slow uptake of innovative techniques
The accepted use in other industries should help to demonstrate that the use of MR in nuclear manufacturing is less innovative now than it was a decade ago. The use in medicine shows that it can function in a heavily regulated environment. Ultimately it is down to the correct use case, if the value is significant enough, the argument to adopt it will be more persuasive.

Documentation
Any use of MR runs through a computer system. This allows for the entire process, or parts of it, to be recorded or logged. In effect, this creates an audit trail to show, what, how, when and by whom it was done. Depending on where MR is used will dictate whether it proves more successful in training or at carrying out a real world task.

Long lead times
MR has been proven to reduce the cycle time in design. DES also has proven benefits, but there is little evidence that MR alone can reduce the overall lead time.

Cyber Security
Cyber security is extremely important in nuclear manufacturing due to the sensitive nature of the manufacturing processes and the Internet Protocol address (IP) surrounding so many elements of it. It is all the more important when considering that manufacturing machines and how they interact with each is being done by wireless systems ever more frequently. The issue of cyber security and the risks there within are covered in more depth in Chapter 5. The chapter is better placed to discuss the issues being focussed on both the 4^{th} Industrial Revolution (IR4.0) and Cyber Physical Systems (CPSs) themselves.

Despite development that spans over five decades (Section 4.1.4) MR can be seen as an emerging technology, as it is not widely used in any industry or yet by the general public. Within nuclear manufacture there is not a single use case attributed to it for design, and very few in other areas. The work that has been conducted is limited, and relies heavily on research done outside MR, drawing from other areas of simulation like DES, Human Factors, Assembly, CAD and prototyping. This makes assessment of the quality of the work done in the field hard to address. The Research Engineer (RE) found a single comparative study of work done in MR (STRASSBURGER et al., 2005), which also contained some of the very few follow up studies. The conclusion of these studies was that MR is a proven technology for taking simulation data that has translated complex data sets into information. This can be extended to the idea being developed in this thesis that combining these information sets in a virtual environment aids understanding, resulting in a gain in knowledge.

The coupling of DES with IVR is one of the only areas where quantitative results are given for the work. The majority of papers are focused on whether it is possible to achieve something in a VE, rather than reasons for developing it. What value or benefit is there is creating this VE? Work by

Christian et al. stands out from the majority of the literature as an example of using quantitative data to demonstrate the effect of using their assembly technique (CHRISTIAND et al., 2007).

To develop MR for manufacturing, it is vital that the scope of the project is defined; parameters for success and failure are identified, and that projects are assessed against this framework. The literature is full of projects that demonstrate what is probable, but what requires development is a method of finding out why, and with what results, developing or using the emerging technologies brings. If the projects are to succeed, success must first be defined.

It is more than possible that other industries are using MR and have duly defined success, but it has to be assumed that this work has not been published in academia. However this does not mean that technology transfer cannot take place. Ming et al conducted research that revealed that upon discovery that 80% of non-conformance issues were the result of assembly, the car manufacturer Toyota started using a system called V-COMM Mockup for assembly (LEU et al., 2013). This has led to well defined results such as:

- 33% reduction in lead time
- 33% reduced design variations
- 50% reduced product development costs

They also found that Toyota are not alone in the automotive industry, as Ford Motors uses Jack Techonomatix and EVart for motion capture analysis, and for analysing the ergonomics of workstations. This has led to the qualitative result of reducing worker injuries, and quantitative result of improving the measured quality of vehicles by 11% (LEU et al., 2013).

Other uses of VR systems by large industrial companies include:

- Airbus use VR on the A350 XWB design and development cycle, and extended it to train workers on manufacture of the XWB A350 (AIRBUS, 2013)
- Boeing used VR when reviewing the design and styling of the 777 (MIZELL, 1997)
- Rolls-Royce Blue Ocean team used an VR maritime bridge for design review and evaluation (BLOOMBERG, 2014)
- BAE systems use VR for the design and work schedule review of 3 Astute class submarines, as well as the type 26 global combat ship (VIRTALIS, 2014a)
- Jaguar Land Rover use VR for design review, most notably the Evoque range (JAGUARLANDROVER, 2011)
- Case New Holland use VR for design milestone reviews, market research, and marketting (VIRTALIS, 2014b)
- Balfour Beatty use VR Engineering for visualisation, training and sensibility studies (BIMPLUS, 2014)
- Acciona use VR in conjunction with BIM for design review (ENCORD, 2014)

- Vestas wind turbine company use VR for design review, maintenance studies, and customer demonstrations (VESTAS, 2014)
- The British Geological survey use VR to view subsurface structures (BRITISH GEOLOGICAL SURVEY (BGS), 2014)

Outside of assembly and assembly training very little research has been done in using MR within manufacturing. This poses the question of why has more work not been done on machining, product transport and other processes? What's more, aside from the Pan et al. (PAN, 2005), studies have not included assessments on the value of using MR in industry. Within the nuclear manufacturing and operation industry there are only qualitative assumptions in the literature on what benefits MR has to offer. An assessment on how effective, value adding, cost reducing, and/or any other benefit the use of MR has on the nuclear manufacture and operation industry needs to be done.

One of the main challenges facing nuclear manufacturing in the UK is that we have not built a nuclear plant in over twenty years. The current plans to build new nuclear reactors will mean a rapid ramping of nuclear manufacturing capabilities, this could be expected to continue as plans to build SMRs progress as well. A system such as the DAAC could help train an inexperienced work force to build reactor components. The use of a wing skin in the project is solely representative of a medium sized work piece. It could easily be swapped out for a reactor pressure vessel, or a steam generator. In such a case the assembly processes could involve bundle building or baffle plate installation.

If the challenges posed by high value, and low volume are to be met, then a transfer of the uses of MR from other industries such as Oil & Gas, Automotive and Aerospace industries into Nuclear could prove to be the safest and most effective. Of particular interest could be the Oil & Gas industry, as although there is little documentation of the use of MR in the industry, any that do exist would be more geared towards tackling similar challenges. This is due in part to the similarities in what the two industries manufacture, and their uses.

MR hardware is in development within big companies like Facebooks Oculus VR, Samsung, and Sony. The adaptation of game engines for VR is pushing the development of software. The resurgence of grass roots interest in VR is taking place online, and in peoples own homes. All this combines to indicate that there will be a great pool of skilled personnel who could work within the nuclear industry. Could they be used to help bridge the skills gap discussed in the nuclear industry? One conceivable means would be to make use of expert users required for creating the VE and the simulations contained within it, to train personnel and convey the complex challenges of procedures in the nuclear industry.

If large companies are taking the *VR Renaissance* seriously, then should the nuclear manufacturing industry not be doing so as well? The application of these technologies could prove game changing, but first the uses for MR within manufacturing need to be identified and the benefits assessed. Previously many VR applications have been solutions that lacked challenges, or projects that failed to deliver on their promise. If MR is to succeed as a viable technology within nuclear manufacture and operation, then use cases that provide value to business must be found.

4.3.1 DAAC

The DAAC should stand on par or above the with the latest developments within MR for manufacturing. It should draw from the knowledge garnered from the rich history of VR and more recently MR. It should be a strong attempt to break out of the big four sectors that have dominated the industry up to this point, and should provide a relatively low cost alternative, while taking advantage of the latest technological improvements.

In many ways the DAAC can be seen as a direct descendant of the link later flight simulators. It is designed to create an environment within which professionals can learn and improve. Of course the DAAC is not solely for training, and has a number of improvements over the link later simulators. As it has both a VE environment and information pass through from the VE to the real by both Voice over Internet Protocol (VoIP) and data projection. This allows for information that is both spatially relevant and temporally relevant to what is taken place within the real manufacturing cell to be delivered to the users. Furthermore the users within the cell can manipulate and work with the VE by using the Android tablets, this is a significant improvement upon Ivan Sutherlands *Sword of Damocles* and has the bi-directional functionality that was so sort after within DES. It allows for multiple data sets from a different systems to be incorporated into the VE and converts them into context-sensitive information for consumption by the users.

The systems sits at the cross-roads of five large themes: Data, Computation, Representation, Interaction, Analytics and real-time system information.

Data The DAAC allows for the importation of a number of different types of data sets, it also creates its own data sets as it records the movement of the users within the real cell. It would also be possible to extend the VE so that it could handle even more data sets and types perhaps in a manners similar to the S-Frame discussed in Chapter 6

Computation The DAAC uses a number of different computing systems and software programs, combining the data from all of these into a game engine that creates the VE. A small network and small specialised computers are used to run the tracking software, with a central small server combining all the data of the Kinects into a single global space. The data is culled here so that the network is not overloaded with multiple REST frames for each individual. An extra Kinect is attached to the main VE server itself to provide the cloud point data of a smaller region. Both this data streams are handled by separate software systems within the game engine itself. The host VE server has a number of guest clients depending on how many users have connected the VE either using Android tablets, Desktop PCs or HTC VIVE steam driven VR headsets.

Representation The users within the VE are represented by simple capsules that have a pair of glasses on to represent which direction they are facing and help other users understand what they might be looking at. The Manufacturing cell users on Android Tablets have the same representation so show other users what they too might be looking at. Finally the manufacturing users are also tracked using the Kinects system, and their full body position and posture is shown by using fully PPE clad avatars.

Interaction The ability to control the VE is within the grasp of all end users. The users within the real cell that adjusts work instructions and uses the VoIP system by means of a hand held tablet. The users on a desktop machine, have access to the same access but can also interact with almost all the objects within the VE, these include all assembly parts, laser points, and projector systems. Desktop users interactions are limited to what they can achieve with just a keyboard and a mouse, as such a number of keyboard shortcuts are provided for them. Users with a VR headset have full access to the VE functions just as the desktop users do. In addition though the interaction is one of a more natural movement, as they are able to the twin wands of the HTC Vive to pick up any objects with their hands, and manipulate it almost as if it was held in their real hand.

Analytics and real-time system information The DAAC considers complexity,computational and networking bottle necks and has been designed to be as modular as possible. This should allow for the addition of extra sub-systems that can be used to analyse the data in either real-time or deferred processing. The capture of human movement, the VoIP and the control and actuation of all elements within the VE all occur in near real-time, taking place as soon as the limitation of computation and networking will allow.

The DAAC covers a number of themes, including what is stated above, it is designed to help address the issues facing manufacturers for the nuclear industry, contribute meaningfully to the body of research surrounding MR, and find a place among the raft of improvement being proposed and implemented as part of the IR4.0

CHAPTER 5

The Fourth Industrial Revolution

Since the early 2010's a fourth industrial revolution has been taking place in several western and developing countries (Bosch, 2012). The herald of this revolution has been the introduction of the Internet of Things (IoT) (Henning, 2013; Lopez, 2014; processengineering, 2013; Witchalls et al., 2013), which has allowed a new level of connectivity between machines and computer systems (Acatech, 2016a). The 4^{th} Industrial Revolution (IR4.0) or Industrie 4.0 represents the increase in automation and data exchange within manufacturing technologies. The term encompasses the application of Cyber Physical System (CPS), the IoT, Cloud Computing and Cognitive Computing.

Modern manufacturing will be conducted in structured, modular and smart factories. These will use CPSs in value adding processes, whilst recording the data to the Digital Twins of the product. The CPSs will communicate over the IoT network within the confines of; the smart factory; the company as a whole; the supply chain and with people, all in real time.

The previous paragraph may sound like nonsense, but this chapter will endeavour to step through the concepts of the IR4.0 , CPSs, Digital Twinning, and smart factories. The chapter concludes with the findings that:

- The IR4.0 presents ideal use cases for Mixed Reality (MR), as a means of visualising context-sensitive data.
- The IR4.0 adds requirements for more advanced training methods for the workforce to allow them to keep pace with technological advances.
- CPS can be used to disseminate technology both horizontally along the supply chain, and vertically within a company.
- Digital Twins offer an effective means of preserving knowledge in an industry undergoing rapid change.

5.1 4^{th} Industrial Revolution

The fourth industrial revolution was first heralded by the Germans in 2013 as Acatech (Deutsch Akadamie der Technikwisenscheiften) presented research and recommendations for implementation to the Federal Ministry of Education and Research Development (BMBF) (ACATECH, 2016a). Within the paper multiple recommendations were made to the German government stating what areas of research should be pursued to further advance the German manufacturing industry. Known in Germany as Projekt Industrie 4.0, the essence of Industrie 4.0 is that;

> "Production is connected with high-quality services. With intelligent monitoring and decision-making processes, companies and entire value-added networks are to be managed and optimised in near real-time" (BMBF, 2017)

The first industrial revolution was heralded by the advent of water mills and later steam powered engines. These methods of manufacture were then superseded by the advent of Henry Ford's, and many others, forms of mass production and automation, in the second revolution. The third industrial revolution was driven by the creation and later advancement of computers, this led to IT systems and robotics being used throughout industry. The fourth industrial revolution is coming about by the miniaturisation of processors and sensors driven primarily by military hardware and the mobile phone industry (KAGERMANN et al., 2013a).

> "While industry 3.0 flourished in automation of single machines and processes, Industrie 4.0 focuses on the end to end digitisation of all physical assets and integration into the digital ecosystem"(COOPERS, 2016)

In one case the implementation of CPS has given rise to a 20% increase in production(BOSCHREXROTH, 2014). Although a great deal of the research and development of Industrie 4.0 has taken place in Germany, the scale of the revolution is now certainly global. A 2016 PWC survey (COOPERS, 2016) found that companies that had moved to implementing Industrie 4.0 technologies had seen digital revenue gains in excess of $490bn p.a and cost and efficiency gains of $421bn p.a. It is hardly surpising then that their key findings included:

- Industry 4.0 is moving from talk to action
- Digitisation is creating leaps in performance
- There are deepening digital relationships with a more empowered customer base
- Focusing on people and culture is the best way to drive the transformation
- Data analytics and digital trust are the foundation of Industrie 4.0
- Robust enterprise-wide data analytic capabilities require significant change
- Industry 4.0 is accelerating globalisation, but with distinct regional variations
- Big investments with big impacts, its time to commit to Industrie 4.0

So what kind of changes are these companies making to see such incredible increases in revenue? A Bosch factory in Germany making ABS/ESP brake systems was an early adopter of Industrie 4.0. The factory added automated warnings from the machines that were triggered before noticeable productivity loses are incurred. The IoT connected machines sent SMS and email messages to workers. These were in turn linked with repair instructions and video guides that are either linked in the messages or arrive with them attached. Employees are provided with an easy to use interface via tablets that allow them to stay "*always connected*" and feed in new solutions and information into the system itself. The move has also allowed further progress towards a paperless production model.(BOSCHREXROTH, 2015)

> "We are using Industry 4.0 to sustainably increase our productivity and remain competitive in the long term" Ruper Heollbacker, Technical Manager(BOSCHREXROTH, 2014)

A recent report from the Advanced Manufacturing Research Centre (AMRC) states that there are three main components to an Industrie 4.0 (AIDEN LOCKWOOD, 2017) system. These are Data, Connectivity and Analytics.

Data - The capture of data is a cornerstone of the IR4.0. Without machines generating data, data driven decision making processes become impossible.

Connectivity - The machines generating data must be connected to each other and/or a wider network. This is why the IoT is considered one of the key drivers of the IR4.0. The connectivity allows for the collection of data sets from multiple machines which can in turn be used in analytics

Analytics - Without analysis then most data sets are effectively useless. The analysis of the data sets informs decision making, which can take place from the machine level all the way up to levels of management.

As seen from the Bosch example, adoption of Industrie 4.0 need not be a fully interconnected autonomously functioning factory floor where data from one machine drives event driven decisions made on the next. Even a simple notification system that provides the factory floor workers with the information they need, when they need it, has presented extensive benefits. This is an example of the use of digital work instructions increasing the collaboration between shop floor workers and their machines. The Bosch example could easily include the ability for shop floor workers to edit the work instructions or supply changes when improved methods were discovered. The increase in collaboration would then be extended to collaboration between shop floor workers. These improvements could then be sent out internally to other factories and externally out across the supply chain. The gains from a single process could be shared and thus multiplied across all similar processes.

Horizontal integration along the supply chain is something that was first pioneered by Toyota and is a central tenant to lean manufacturing (WOMACK et al., 1990). The advent of the IR4.0 has allowed for an even greater level of integration and sharing of data. The sharing of best practices along the value chain can multiply the gains made within a single process. One area that this is often highlighted is in Design where collaboration between manufacturers can be essential. This is especially true in Design for Manufacture and Assembly (DfMA).

Horizontal integration allows for the reduction of value-added operations within a single factory, through the use of collaborative manufacturing (H. W. LIN et al., 2012) and collaborative development environments (MENDIKOA et al., 2008). The collaborative network spreads and balances the risk and allows for a wider range of market opportunities through the sharing and combining of resources (CHIEN et al., 2013). Other benefits can include higher agility, faster adaption to volatile markets and a shorter product life-cycle. A commonly cited challenge associated with increased horizontal integration is the spatial separation and numerous networks due to multiple production sites (JAEHNE et al., 2009) as well as the, often less than efficient, communication across company departments and boundaries (DAVIES et al., 1998). These challenges only increase the need for greater levels of coordination and integration of the production data. In the supply chain agility, goes hand in hand with the ability to track commodity flaws and data delivery (BRETTEL et al., 2014). Central data pools can be used to collect a huge array of sensor information from across the production network. This can then be made accessible and increase the communication across the value chain.

With high levels of horizontal integration it is possible for a company to move from making a product into selling a service. Such a network of businesses are often referred to as virtual corporations. The dispersed network idea means companies will have to focus on more Business to Business (B2B) trade. This should be done by focussing on core competencies and out sourcing other activities to the network (CHRISTOPHER, 2000). Traditionally this increases the risk associated with an individual company as it can become a provider of only a single service. If that service were to be superseded or become redundant then the business would also become so.

End to end digital integration will allow for a degree of virtualisation, which will enable new levels of remote assistance and updates through embedded devices. This will drive an increase in value added services for products post initial sale. As a secondary effect, it will also create large data sets of either user or system information that can be harvested for later development and production cycles. This is an ideal use case for Virtual Reality (VR) and Augmented Reality (AR) as it is good for context-sensitive visualisation of data (BRETTEL et al., 2014)(PAELKE, 2014).

The reduction of innovation and development periods are helping to increase the high innovation capabilities of companies. Which reduces the *"Time to Market"* in an environment where customisation is in high demand (BALDWIN et al., 2000). This increased flexibility is allowing buyers to refine the conditions of trade to ever smaller batch sizes; forcing companies to become ever more agile(HEINER LASI et al., 2014). One way in which they are achieving this is through decentralisation of the decision making process and mass customisation (DA CUNHA et al., 2010; FOGLIATTO et al., 2012; KHALAF et al., 2010)

The real-time monitoring of machines can also add to the quality chain itself. Early fault identification can lead to predictive maintenance of machines which reduces both machine downtime and quality issues. The increased use and reliance on sensors is allowing for greater versatility and automation in manufacturing (LUCKE et al., 2008), which has led to the creation of CPS and Smart Factories.

5.1.1 Challenges

Horizontal and vertical digital integration, creates a holistic approach to data and systems within a company. However many of the systems, processes and infrastructures of these companies were designed in isolation. The challenges for companies wishing to adopt Industrie 4.0 technologies include:

Standardisation & IT Architecture - As the complexity of IT systems continue to grow in industry, the requirement for management of those systems also increases. To realise their potential as powerful control systems, they must be reliable and interoperable. Uniform standards are needed across industries (TAO, L. ZHANG et al., 2011), especially the supply chain (KHALAF et al., 2011), and manufacturing capabilities (H. A. ELMARAGHY et al., 2012). This would also reduce the barrier to entry for Small to Medium Enterprises (SMEs). It is the opinion of the Research Engineer (RE) that the requirement for standardisation across systems within the sphere of IoT and CPS is so high that it should not be left to competition between companies that provide the software but instead be governed by a regulating body in much the same way that internet standards are. The requirement for systems to be able to communicated effectively and without the differences of proprietary software preventing full integration between systems should be a high priority. It would both reduce the likelihood of error and help reduce the complexity of translating one system to interface with another. A separate company independent task force or regulatory commission could look at what is the best solution, rather than what will prevent a companies competitor from gaining a foothold in the market.

IT Security - As the data generated and broadcast around a factory increases, the possibility of data corruption, loss or espionage also increases. Securing the IT systems are essential to removing system errors either forced or accidental.

Qualifications - Adaption of training and qualification of the existing skilled and unskilled workforce is essential to allow them to move forward with the new manufacturing concepts and technologies.

The biggest challenge is people themselves, such as the Digital leaders, CEOs, CTOs and CIOs. This includes communicating the Industrie 4.0 transformations down through digitally qualified staff for implementation on the shop floor. Radical disruption isn't always comfortable so how the change management is handled becomes critical. Enhancing the organisational structures and skill sets will be vital to a successful transformation.(COOPERS, 2016)

In this regard qualification becomes a key success factor to avoid a double digital divide - between large and small industrial enterprises on the one hand, and between high and low-skilled workers on the other. Based on an empirical survey, the Acatech POSITION paper (ACATECH, 2016b) analysed the competencies required for both SMEs and larger companies to adapt to the IR4.0. The digital divide is in people as well as technology. Training staff on and in CPS is essential especially at the SME level. The interconnected nature of Industrie 4.0 places new demands on employees and management, but also provides for the possibility of highly tailored training and professional development. This increases the potential for both functional and cross skilled employees. The ability to understand and work in collaboration with the CPS of Industrie 4.0 will prove essential to both adaptation and adoption.

5.2 Cyber Physical System (CPS)

The IR4.0 is built upon the creation of interconnected systems that can communicate with each other and the work force.

> "Cyber-Physical Systems (CPS) are integrations of computation and physical processes. Embedded computers and networks monitor and control the physical processes, usually with feedback loops where physical processes affect computations and vice versa." (E. A. Lee, 2008)

These Cyber Physical Systems (CPS) are the back bone of Industrie 4.0 and allow for increased levels of optimisation (HS Yan et al., 2007), mass customisation (Da Cunha et al., 2010; Fogliatto et al., 2012; Khalaf et al., 2010), virtualisation (Brecher et al., 2011), modularisation (Qiao et al., 2006; Haiping Zhu et al., 2010) and Production Planning & Control (PP&C) (El Haouzi et al., 2008; Lian et al., 2007). This is a tall order from something often defined as emerging systems, where physical and digital representations cannot be differentiated in a reasonable way (Heiner Lasi et al., 2014).

CPSs are automated systems that enable connection of the operations of the physical reality with computing and communications infrastructure(Acatech, 2016a; Radhakisan Baheti et al., 2011; E. A. Lee, 2008). Unlike embedded systems, the focus is on networking (E. A. Lee, 2008; Riedl et al., 2014; J. Wan, Hehua Yan et al., 2013) and communication, most likely through the IoT. The IoT can be loosely defined as a set of network objects that share data and information. IoT physical objects can be integrated into the information network, where they can become part of the CPS or Cyber Physical Production System (CPPS). This allows for horizontal data integration through the value networks, alongside vertically integrated networked manufacturing systems; with the goal of end-to-end, top-to-bottom digital integration of engineering, across the value chain(Acatech, 2016b). In this way CPSs could be used as part of a global value creation network (Frazzon et al., 2013).

To achieve this "information from all related perspectives is closely monitored and synchronised between the physical factory floor and the cyber computational space" (Jay Lee, Bagheri et al., 2015). CPSs, therefore, fulfil a variety of different functions, including but not limited to; managing the interconnected systems between the physical and computational (R. Baheti et al., 2011); expanding the capabilities of the physical world through computation (Radhakisan Baheti et al., 2011); generating big data within factories via a growing array of sensors and networked machines (Shi et al., 2011)(Jay Lee, Lapira, Bagheri et al., 2013) and managing big data through connected machines that are; intelligent, resilient and self adaptable (Krogh, 2008) (NIST, 2013). The expectation of an Industrie 4.0 factory is that the CPSs will generate significant economic benefit (Lee et al., 2013)(Jay Lee, Lapira, S. Yang et al., 2013).

CPSs are made up of two main components (JAY LEE, BAGHERI et al., 2015):

1. Advanced connectivity that ensures near real-time data acquisition from the physical world, and information feedback from user-space
2. Intelligent management, analytics and computational capability that constructs the cyber space

Which should benefit a company through enhanced; equipment efficiency, reliability and product quality.

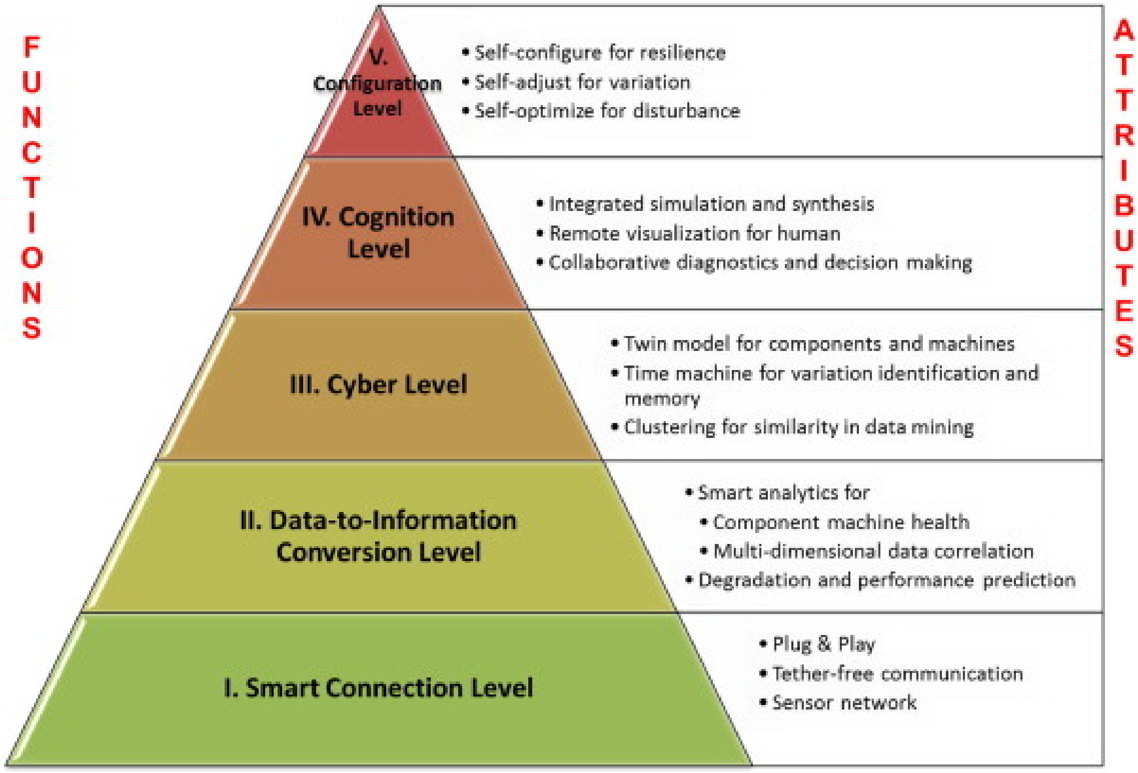


Figure 5.1: 5C architecture for implementation of Cyber-Physical System (JAY LEE, BAGHERI et al., 2015)

CPSs are systems that depend on the synergy of cyber and physical systems (E. A. LEE, 2010) making them difficult to analyse, design and validate(ZHANG et al., 2013). As Zhang et al put it:

- CPS must account for the interaction of interdependent behaviours of cyber & physical components to achieve system level functionality
- CPS conjoins; continuous dynamics of the physical components, and discrete dynamics of cyber components

A hierarchy of integration created by Lee et al (E. A. LEE, 2008) shows the benefits of embedding CPSs into a system wide network (Figure 5.1). Each level of integration provides greater potential benefits, improves the use of the data systems until the self-adjusting, self-optimising, self-configuring CPS is created. The progression of data up the pyramid represents the conversion of data into information which was first discussed in Section 3.3.

These are not the least of the design challenges that companies face when wishing to create CPS for industry.

> "To realise the full potential of CPS we will have to re-build computing and network abstractions. These abstractions will have to embrace physical dynamics and computation in a unified way"(E. A. Lee, 2008)

As it stands, most network components are designed for parsing information in the format and language that other computers are designed to interpret. This is computers communicating with each other in a language designed for them. Yet many programs fail, crash, segfault, disconnect, and/or return erroneous information while communicating with each other. What is being attempted with CPSs is computers talking computer languages, about the physical world, to other computers. This change is important enough to require a set of Hardware Abstraction Layers (HALs) for converting real world information into computer readable formats, which can then be exchanged with other computers. Another set of HALs are then required to interpret that information back into a format that the physical (shop floor machine, human being or other end user) can interpret and parse itself in real time.

Those last three words mean something very different in Information & Communications Technology (ICT) than they do in the physical world.

> "In the physical world, the passage of time is inexorable and concurrency is intrinsic. Neither of these properties is present in todays computing and network abstractions"(E. A. Lee, 2008)

Real-time in ICT is as fast as the processor can handle the data; the computer is processing the information without storing it, analysing it, and then optimally processing. Instead real-time is the processor analysing the data as it comes, which means it inevitably backs up, leading to an unknown period of delay before finishing the operation. In the physical world, real time means this exact moment. For example the machine is cutting the metal right now, or if the critical temperature at which the machine should increase cooling flow has been reached, then this action should happen immediately, not once the on-board system or other processor has finished working through the information. Time can not be abstracted away and must be raised throughout the abstraction layers to ensure better real time capabilities. The analogy in manufacturing is that the factory must run to the beat of the drum of the slowest part.

Such an issue can be compensated for in multiple ways; using other levels of abstraction, reducing the use of multiple threads or using dedicated sub-systems for mission critical hardware. However, for safety systems, at least within the nuclear industry, these sub-systems often have to be isolated. Components at any abstraction level must be predictable and reliable if a technology is to be feasible.

> "The simplest C program is not predictable and reliable in the context of CPS because...."(E. A. Lee, 2008)

As we move further up the abstraction layers, for example to an operating system (OS), we move further from perfect predictability and reliability to wildly non-deterministic behaviour that must be managed by the software designer (E. A. LEE, 2006). However, such abstraction layers are essential to ensure cross domain homogenisation of the systems (SZTIPANOVITS et al., 2012). Through standardisation of the abstractions and architecture, it should become possible to start a modular approach to CPS (RADHAKISAN BAHETI et al., 2011). However, standardisation is one of the largest hurdles standing in the way of more widespread adoption of CPS (ACATECH, 2016b; AIDEN LOCKWOOD, 2017; RADHAKISAN BAHETI et al., 2011; JAZDI, 2014; E. A. LEE, 2008; SZTIPANOVITS et al., 2012). This is especially true with respect to data models and exchange formats (LASI et al., 2014). Once Standardisation has been reached, even if just across a single value chain, then it would allow CPSs to span an entire supply chain (Figure 5.2.

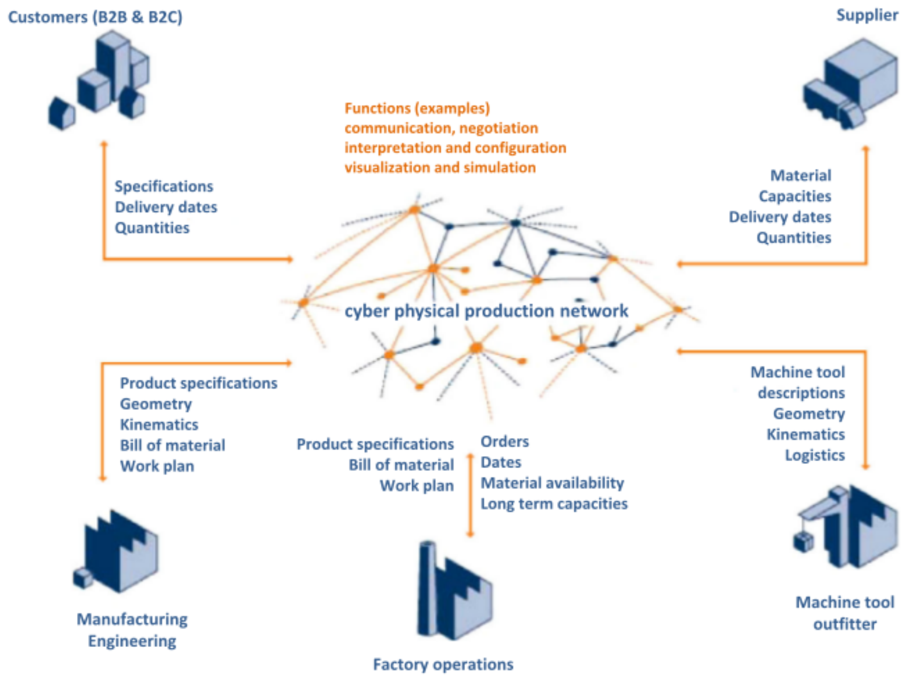


Figure 5.2: Interdependent supply chain (GEISBERGER et al., 2012), p56

The increased levels of communications across systems could allow for:

- New methods of modelling (FETTKE et al., 2004)
- Innovative Manufacturing Executions System (MES)/Enterprise Resource Planning (ERP) manufacturing execution systems (KLÖPPER et al., 2012) (KOCH et al., 2010)
- Digital product memories (BRANDHERM et al., 2011)
- Innovative platforms architectures (WAHLSTER, 2014)

The two other main challenges would be reliability and security (ACATECH, 2016b; AIDEN LOCKWOOD, 2017; COOPERS, 2016). Reliability is an issue because of the potential consequences of failure, which could range from inconvenient to catastrophic, depending on the real world system itself. Security of the systems and the data they parse are a major concern as well; intercepted data could be used in many forms of industrial espionage. If Israel is capable of shutting down the uranium enrichment program of Iran through the use of the Stuxnet worm (N. ANDERSON, 2012), then it would seem probable that governments, or other parties, may be able to do the same to any manufacturing system.

Advances in ICT allow measurements of product data from production status, material condition and location, to work instructions. With the use of Radio-Frequency Identification (RFID), this data can be associated with individual products. This allows for the control of product manufacture processes at a much higher resolution than previously possible (BROSZE et al., 2007). The creation of digital ghosts that represent a product, either at the individual or generic level, is called Digital Twinning. However, a Digital Twin should not be considered as limited to the product that is being created in the value chain; just as all items within a factory can be considered a product from one value chain or another, so can they all have Digital Twins.

5.3 Manufacturing Ontology

If the future of manufacturing is dependant upon systems interacting and understanding each other, then it is of paramount importance that the people who are writing these systems and working with each other also share the same level of understanding. This is as true in manufacturing as it is anywhere else. If one engineer is used to measuring everything in inches, and the other in millimeters then the transferral of plans may not result in the same product being made by each engineer. This can be easily extended to common parlance, currently in manufacture a number of ontological systems already exist, such as Kanban, Just In Time (JIT), Lean manufacturing, 7 wastes and many more. These systems all apply a common language and explicit definitions.

In manufacturing simply pushing information and actions is insufficient to transfer production standards, this is especially true within global manufacturing (ZHOU et al., 2004). A shared manufacturing knowledge and understanding needs to be created to bridge between different groups of people, computers and networked machines. With the increasing dependence on technology within manufacturing it is becoming ever more important that a shared knowledge and understanding is defined (GÓMEZ-PÉREZ et al., 2006). How else could systems running in Japan, update American stock levels on completion of manufacturing processes? Product Data Management (PDM) or ERP or MES often contain disparate or scattered information about processes and control. Any overarching system will need to find a way to interface with all of these systems using a common language. The issue is one of syntax that allows computer systems to communicate effectively and a semantic problem that is faced by users dealing with the conceptual issues that surround the information (PANETTO et al., 2012). A good system require reference standards, that allow for effective communication of concepts and ideas between groups. This is not dissimilar to the standardisation programs that are used to govern the internet, and computer networking.

One of the proposed solutions to these issues is the creation of a manufacturing ontology. It seems odd that something based around the definition of terms has in itself attracted so many of its own definitions:

> "an ontology defines the basic terms and relations comprising the vocabulary of a topic area as well as the rules for combining terms and relations to define extensions to the vocabulary"(NECHES et al., 1991)

or

> "An ontology is an explicit specification of a conceptualisation"(GRUBER, 1993)

or

> "ontologies are defined as a formal specification of a shared conceptualization"(BORST et al., 1997)

or

> "a set of logical axioms designed to account for the intended meaning of a vocabulary"(GÓMEZ-PÉREZ et al., 2006)

Oddly it is the earlier definition from Neches et al. that makes the most common sense, that is in part due to the formal specifications that have entered into the academic sphere of ontology since then. As with many topics, the more specialised and detailed the it becomes over time, then the more exacting and carefully defined the language surrounding it becomes.

More specifically to manufacturing ontologies it has been established that there are three main streams of language that are used when discussing manufacturing techniques. These are the flow of material, the flow of information, and the flow of cost. The unification and harmonisation of these three streams are vital for efficient manufacturing activities (HITOMI, 2017) and thus a formalisation of the language being used when discussing these streams is also vital.

Information flow involves aspects as varied as production planning and control, Material Resource Planning (MRP), ERP, and Factory Design & Layout. Information flow can be further classified into technology information flow and management information flow. The technology flow consists of all activities and information pertaining to actual design and manufacture. Whereas the management information flow consists of the sales plan, production planning, and scheduling activities. Thus for the purpose of the Digitally Assisted Assembly Cell (DAAC) the technology flow is of greatest interest.

Manufacturing activities always create product properties of shape, place, time, cost and value (HITOMI, 2017). It is the actors that execute these activities, in manufacturing these actors are humans ranging from managers, engineers and shopfloor operators. The tasks are performed using methods, these are clearly stated processes which help free up the actors mental work load, and assist in increasing the efficiency of the actors. They take the form of rules, codes and policies, all to help guide the actors to executing the activities. By tracking and gathering data on these activities it is possible for people or algorithms to optimise the results to increase transfer speeds, set the best routing through a factory, or ensure the correct amount of materials are in the right place at the right time (ZHOU et al., 2004). These results are tangible and explicit, and it is hoped that the DAAC can help by increasing the amount of data available through human tracking. Thus allowing further levels of optimisation.

There are aspects of these activities which are not explicit, but instead implicit. Implicit manufacturing knowledge is that which is hard to define in words, or explain to others. An example is the knowledge that exists in an individuals brain, and a groups value, experience, know-how, culture and motions. It is these motions that the DAAC also seeks to capture, not only for optimisation with the same facility, but for export to other manufacturing and training environments, and to store for future generations.

5.4 Digital Twin

The concept of the Digital Twin was introduced in a talk about Product Life-cycle Management (PLM) by Michael Grieves in 2003 at the University of Michigan (GRIEVES, 2014). The term was later coined by John Vickers of NASA(GRIEVES, 2011). A Digital Twin is just the digital representation of any product. This can be generic, similar to idea of Platos' forms (VLASTOS, 1980), or specific to an individual component or product.

A Digital Twin consists of three main parts (GRIEVES, 2014):

- Physical products in real space
- Virtual Products in virtual space
- The connections of data and information that ties the virtual and real together

The data and information that ties the virtual and the real together can include as many sources as required. Data included can either be meta-data inherited from previous processes or the creation of the Digital Twin or data can be drawn from MES on the shop floor such, as but not limited to:

- Non-Destructive Testing (NDT) sensors
- Gauges
- CMM
- Laser measurement
- Vision systems
- White light scanning

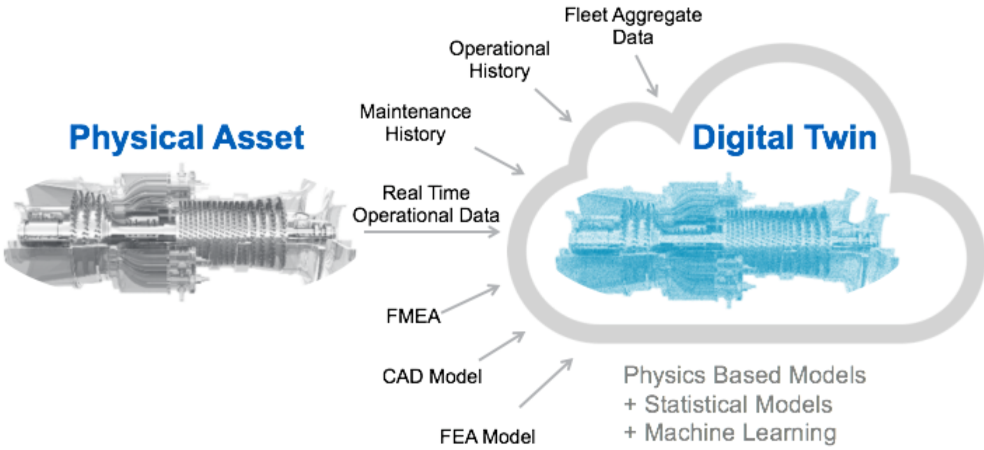


Figure 5.3: Representation of a gas turbine and its Digital Twin (GE, 2018)

The benefit of such a Digital Twin is that it can allow items to be tracked through their production and subsequent testing. The information can then be used in both Quality Assurance or future design iterations and modifications. For example, an aircraft could be tracked by its tail number(TUEGEL et al., 2011), the Digital Twin could then be used to store all the:

- Thermal heat transfer model
- Dynamics model
- stress analysis model
- fatigue model
- Computer Aided Design (CAD)
- Computational Fluid Dynamics (CFD)
- Materials meta data

The data can be gathered and stored as the plane and its constituent parts are passed through different engineering teams. In an industry that relies on engineering judgement, which is accrued from years of experience, this can be easily lost in retirement etc. When the design and build of products can span decades this poses an even greater challenge.

Within a design cycle, a previous product's Digital Twin can be used within simulation. The benefit of this is that it allows the use of the Digital Twin's meta data variables to both stimulate and bench mark against the simulation outcomes. This can assist in the prediction of possible failure modes and could ultimately be used to help in management and certification of products. In this way, the digital memories of the object can be used to improve manufacturing techniques (BRANDHERM et al., 2011). This can be extended into what are referred to as Digital Passports, where the Digital Twin acts is a certifiable product that can be used to verify the history of manufacture of an object

(AIDEN LOCKWOOD, 2017).

A Digital Twin can be more than just a repository for data, it can be an analytical tool in itself. If the Digital Twin were to be an engine, then it could be collecting and analysing sensor data at the same time. The results could then be fed back to CPSs and used to schedule future maintenance of affected areas, or to demonstrate the performance of a new coolant or other element. In this way, and others, the Digital Twin can form a key part of the Smart Factory.

5.5 Smart Factory

A Smart Factory is the terminology used for an Industrie 4.0 enabled factory. This includes the use of IoT, CPSs and Digital Twins, to create a live, adapting factory that uses its interconnected systems to allow for both optimisation and variation of systems, processes and product. One such example of this is the Reconfigurable Manufacturing Systems (RMS)(ABELE et al., 2007) used to adapt factories to changes in production requirements in a more cost effective manner than traditional production lines. These allow manufacturers to change machines, people and components, as and when required. A comparison of the technical features of a Smart and Traditional factory can be found in Table 5.1.

The IoT works as the back bone for data transfer within a Smart Factory (F. CHEN et al., 2015; JING et al., 2014; TAO, ZUO et al., 2014). The use of Wireless Sensor Networks (QIU, XUE et al., 2006) (QIU and SHA, 2007) provide a raft of Big Data sets (M. CHEN et al., 2014) from smart data generating systems. The data can then be stored remotely in Cloud Computing systems (XUN XU, 2012) (Q. LIU et al., 2014) (J. WAN, D. ZHANG et al., 2014), which can be used to analyse the data and feedback into the Cyber Physical and Embedded systems (J.-F. WAN et al., 2010) within the Smart Factory, across the value chain, and within the company. This allows for industrial production that is:

- Highly flexible in production volume
- Highly customizable
- Extensively integrated with customers especially B2B
- Sustainable (SHROUF, ORDIERES et al., 2014)

Smart Factories (Figure 5.4) rely on the IoT and Smart Objects, that is, objects that accrue data through sensors sets which then form a part of the CPSs (MIRAGLIOTTA, PEREGO et al., 2012). The advancements of CPS and CPPS has allowed for distributed and interconnected production facilities. These Smart Factories record information data, derive information and adapt to it. The knowledge is then stored internally, in the company or in the cloud. This allows Smart factories to change orders, self optimise and self reconfigure as demand suits (FESTO, 2013). In this way Smart Factories can be seen as living entities that adapt in real-time as the data, information and knowledge changes (AZEVEDO et al., 2011).

Living and working within these smart factories are augmented operators. These are workers that are connected into the network of information, with the ability to respond to any issues or changes that take place in real time. The augmentation can take many different forms, from the basic level that we see in modern supermarkets where all the staff have headsets and can communicate using Voice over Internet Protocol (VoIP) systems, all the way up to wearable MR devices that will provide full scale data overlays onto the real world objects. The most common augmentation however would probably through the use of a smart tablet or device. This device would effectively replace the clipboard as a storage device for what is taking place within the factory. The smart element will become important as it could pull this data from servers within the facility, and allow for changes to the data to be pushed back. In this way an operator could be informed that a manufacturing run is coming to an end, and that they are required to set up for a new one. This could involve a message being sent to the operator, and then a set of directions to where they need to reset which machines for a new run. The tablet could be hooked up to a factory floor plan and provide directions as to where the operator needs to go, it could also be hooked up to the Material Resource Planning (MRP) system showing the operator how much stock material is at the required stations, and how much more must be retrieved. The augmented operator is about giving staff on the factory floor access to all the information they need, when they need it. This includes being smart with what information is available, as there is no point in showing the operator a full weeks worth of work instructions each time they want to complete a days task. Instead they should be shown just the work instructions they need for that moment, and the extra space should be filled with context sensitive data surrounding those tasks.

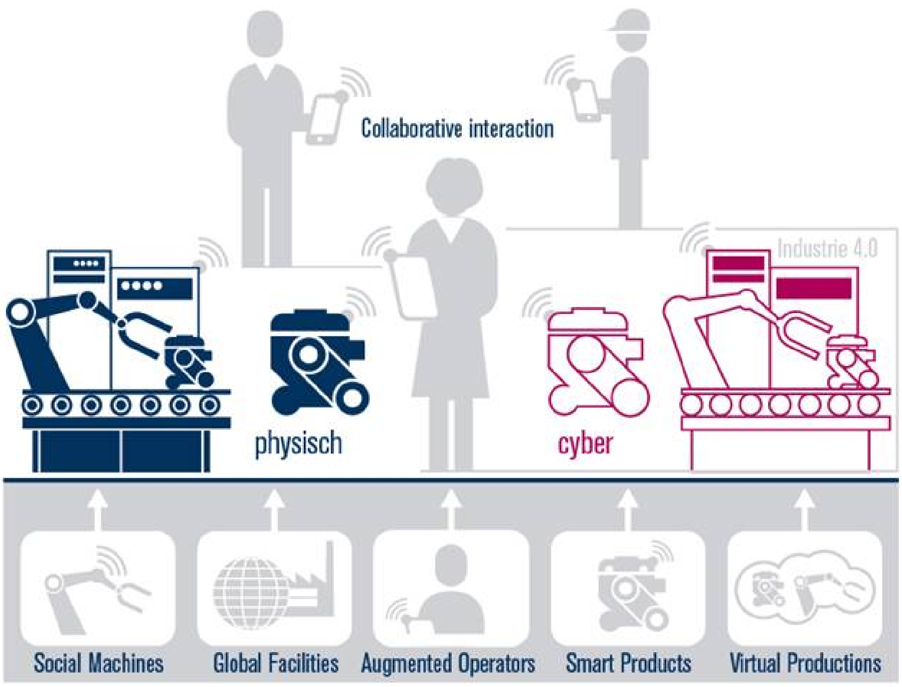


Figure 5.4: Collaborative interactions in a Smart Factory (PROF. DR. HENNING KAGERMANN, 2012)

Smart Factory Characteristics:

- Mass Customisation (FOGLIATTO et al., 2012)
- Flexibility
- Factory visibility and optimised decision making (PROCESSENGINEERING, 2013)
- New planning methods for factories (ZUEHLKE, 2010) (SHROUF, ORDIERES-MERE et al., 2014)
- Creating values from big data
- Creating new services
- Remote monitoring
- Automation & a change of role for workers
- Pro-active maintenance (LOPEZ, 2014)
- Connected Supply chain (LOPEZ, 2014)
- Energy management HVAC (Heating Ventilation, Air Conditioning) (HALLER et al., 2008) (VIKHOREV et al., 2013) (MIRAGLIOTTA and SHROUF, 2012)

A key element of Smart Factories is the idea of modularisation (RADHAKISAN BAHETI et al., 2011; KAGERMANN et al., 2013b). Separating the factory into units that share the required information will reduce the complexity of coordination and increase flexibility. This modularisation can lead to a focus on smaller optimisation and in consequence the neglect of larger optimisations at greater levels or scale. An overarching value model can be used to combine the two paradigms; as sub-optimal solutions in one cell could resolve the bottle neck in another. To this end, the optimisations should still adhere to the beat of the drum across the modules.

One such example of an overarching value model is Delone & McLean Information Success (IS) model (DELONE et al., 2003). The model uses a set of six critical dimensions of success: information quality, system quality, service quality, system use/usage intentions, User satisfaction, and Net system benefits. The system has been widely used in the last few decades to asses how effective ERP implementations within companies have been (FAN et al., 2006; WEI et al., 2009). At Warwick university the model has even been used to assess VR learning environments (ACTON et al., 2009).

The methods that all this information is shared amongst machines and systems, is a part of the large security conundrum that publishing all this data to a *cyberspace* creates. Traditionally solid ethernet wires are used to connect machines and computers together within a factory. This has the advantages that non registered users can not connect to you network without compromising the physical security of your manufacturing space. It also requires that space for routing all these cables into network switches is created. This is manageable when you are designing a factory, but can be difficult to implement in an existing factory that was designed pre-networked devices. Furthermore some manufacturing process are not conducive to having thin metal and plastic cables anywhere near them. Large scale welding, heat treatment, and chemical treatments can all cause serious damage to hard cable network solutions.

Table 5.1: Technical features of smart factory compared with the traditional factory (S. WANG et al., 2016)

Smart factory production system	Traditional production line
Diverse Resources To produce multiple types of small-lot products, more resources of different types should be able to coexist in the system	**Limited and Predetermined Resources** To build a fixed line for mass production of a special product type, the needed resources are carefully calculated, tailored, and configured to minimize resource redundancy
Dynamic Routing When switching between different types of products, the needed resources and the route to link these resources should be reconfigured automatically and on line	**Fixed Routing** The production line is fixed unless manually reconfigured by people with system power down
Comprehensive Connections The machines, products, information systems, and people are connected and interact with each other through the high speed network infrastructure	**Shop Floor Control Network** The field buses may be used to connect the controller with its slave stations But communication among machines is not necessary
Deep Convergence The smart factory operates in a networked environment where the IWN and the cloud integrate all the physical artifacts and information systems to form the IoT and services	**Separated Layer** The field devices are separated from the upper information systems
Self-Organization The control function distributes to multiple entities These smart entities negotiate with each other to organize themselves to cope with system dynamics	**Independent Control** Every machine is preprogrammed to perform the assigned functions Any malfunction of single device will break the full line
Big Data The smart artefacts can produce massive data, the high bandwidth network can transfer them, and the cloud can process the big data	**Isolated Information** The machine may record its own process information But this information is seldom used by others

With the addition of mobile devices, and especially augmented workers the need for wireless access arises. This has the benefit of being able to be installed into existing facilities, even if getting strong enough signal throughout the facility could be a struggle. This allows both machines and operators to move freely around the facility while maintaining a connection to the network, a WiFi network also allows for positioning data to be created using triangulation between difference connected routers. This means that you have a wealth of secondary data, and can even locate portable machines or operators based solely on their collective signal strengths from different sources. However it does mean you are broadcasting your factory data systems into the ether, this means that physical access is no longer required to gain access to your information systems. This is a clear loss of a layer of security, and create a larger onus on the existing network security systems to be impenetrable, something that has thus far proved impossible even for the most secretive organisations.

A partial solution to the security issues could be Light Fidelity (LiFi), this uses light beams to communicate data in a straight line between devices. It works in a similar manner to fibre optic cables, where a laser or other light source sends out a series of binary on off signals that can then be decoded and converted back into electrical signals on the receiver. There is great potential for LiFi to increase security, as it once again requires physical access as part of the attack vector. It also requires that communicating machines have a clear line of sight at all times they are expected to communicate with each other. It is also impossible for augmented operators to interact with, unless they do so at predefined stationary positions. As such LiFi is probably only part of a future solution for increased data security within manufacturing facilities.

5.6 Conclusion

With a recent study showing that our current manufacturing paradigms are unsustainable (ALKAYA et al., 2015), how far away is industry from adopting Industrie 4.0 and Smart Factories? Out of the manufacturing companies that partook in the Economists Intelligence Unit 2013 Survey: 96% expect their business to adopt IoT and 63% believe slow adaption will damage business.(WITCHALLS et al., 2013). A year later the American Society for Quality (ASQ) found that of companies claiming to have implemented a smart factory 82% increased efficiency, 49% had fewer defect and, 45% increased customer satisfaction (QUALITY, 2014). The 2016 PWC survey found that manufacturing companies are making $907bn p.a in digital investments towards Industrie 4.0 (COOPERS, 2016). It is not just companies that have taken an interest, the revolution and potential advantages have been identified as large enough for nation states to get involved, by investing in research towards wider implementation. The Germans have created the *High Tech strategy for 2020 action plan*(ACATECH, 2016b), The USA has the *Industrial Internet USA*(CONSORTIUM, 2017), China has *Internet+*(STATE COUNCIL OF CHINA et al., 2017). The UK has research funded by the EU 7^{th} Framework and in parts by Horizon 2020 (8^{th} Framework)(2014-2020) in the region of €80 billion. With both private and public interests funding the research and development of Industrie 4.0 technologies, it looks as if the 4^{th} Industrial Revolution (IR4.0) is set to continue.

The analyses of specific competencies within large companies and SMEs has shown that both need to design and adapt existing products and processes, whilst developing new business models. There is a real prospect of what is being called the *"Double Digital Divide"*, separating large companies that can afford the technology to adapt to Industrie 4.0 from SMEs that cannot. The second part of the divide is within people. Larger companies and enterprises are more likely to be able to afford to take staff away from key roles and re-train them. Often it is the lack of standardisation that means the cost of adoption for SMEs is much larger as a percentage of their company (ACATECH, 2016a).

This Chapter has discussed some of the main challenges surrounding the IR4.0 and focussed on key constituent terms and technologies associated with it. The findings can be summarised as:

- VR, AR, and IR4.0 have all been driven by advances made by miniaturisation of processors and sensors propelled by military hardware and the mobile phone industry
- Industrie 4.0 provides an ideal use case for VR and AR as it is optimal for context-sensitive

visualisation of data (Brettel et al., 2014)(Paelke, 2014).

- The training of staff with CPS is essential to the adoption and adaption of modern manufacturing techniques associated with the IR4.0 (Acatech, 2016a)
- That modularisation can lead to a focus on smaller optimisations, to the neglect of larger optimisations at greater levels or scale. An overarching value model can be used to combine the two paradigms as sub-optimal solutions in one cell could resolve the bottle neck in another. To this end, the optimisations should still adhere to the beat of the drum across the modules.

- CPSs offer the potential for communication of technologies and changes horizontally across the value chain, and vertically within the company and as part of a global value chain (Frazzon et al., 2013).
- Digital Twinning is effective as a means of preserving knowledge in an industry that relies on engineering judgement, accrued from years of experience. Knowledge can be easily lost in retirement or other loss of employees. When the design and build of products span decades this can pose an even greater challenge.
- Producers must set priorities and upgrade the Workforce as well as its technology (Rüssmann et al., 2015)

5.7 DAAC

As previously stated the DAAC aims to capture not only the explicit data and activities of the work force, but also the implicit motion, values, and know-how of groups of manufacturers. It is very much a product of and for the IR4.0, it is probably easiest to see the DAAC as a CPS, although it certainly covers other areas covered in this chapter. Whether that be the use of augmented operators, or the interconnected nature of its systems and components, or its ability to allow access to its data streams from any location in the world, or its representation of the real manufacturing cell as a Digital Twin.

The DAAC allows for the extraction of real world data into the virtual for storage and an infinite number of reviews. The data it stores is not only work instructions and manufacturing techniques but the movements of the manufacturers themselves. In this respect it serves as an important bridge between the computer and machine based CPS data and the manufacturers themselves. Currently data systems have concentrated on machining times, product flow, lead time and many others, with little to no data being generated and then used of what the manufacturing operators are doing within the factory. The operators actions where like quality black spots that moved around the web of data and information flow within a factory. The DAAC can help to address this, by filling in some of those voids.

As a Digital Twin the DAAC can also be used as a production planning and optimisation tool prior to any deployment on site. The use of the Virtual Environment (VE) allows for change and testing cycles to happen even before a facility is created. The gains made in these stages can then be conveyed to the work force by means of projection markers and other indicators within the VE, but viewed from within the real manufacturing cell.

The data then generated in a real manufacturing environment can then be stored back into the VE for future testing and verification. Standards of manufacture could be set this way for export to other manufacturing cells or facilities. The VE could also be used to store legacy data, and help to ensure that how processes used to be carried out are also stored for review. Furthermore infrequent operations could be stored in this way, and then rehearsed both in the VE and the real manufacturing cell to ensure that they are carried out correctly, all with the assistance of the DAAC.

Currently the DAAC does not conform to coding or manufacturing standards, this is something that will need to be reviewed with experts from within both areas. It is hoped that further benefits of the cell could be found when this is done.

In the next chapter the RE will discuss a real world Rolls-Royce (RR) project that investigated the potential of harnessing a VE to display data in a context-sensitive manner.

CHAPTER 6

Smart Frame

As an introduction to the use of Virtual Environment (VE) this chapter will detail work completed by the Research Engineer (RE) towards the Rolls-Royce (RR) Smart Frame project. The chapter first introduces the Smart Frame concept, before detailing the REs involvement and objectives. This is followed by a more in depth look into the required methods for importing data sets into Unity3D (Technologies, 2014), and the use of the game engine as a VE. The chapter concludes with a discussion of the merits of the project, and its reception at the company.

The chapter as a whole serves as a case study of how data integration within a VE can be used within a global manufacturing company, and its reception.

6.1 The Rolls-Royce Smart Frame

The RRs Smart Frame was a concept developed under guidance of Martin Goodfellow with the aim to create a modular building technique for advanced Small Modular Reactors (SMRs). The idea was to create the parts of an SMR in blocks the size of shipping containers which could be constructed on a factory floor. The constituent parts of the reactor would be manufactured within these frames, and then transported to site by conventional means. Once on site, the parts could be connected together to form the systems required for a nuclear reactor. This method has many advantages over the current methodology of building reactors but the one relevant to this chapter is the idea of it being a *smart* frame.

The frames were to be designed to be part of a Cyber Physical Systems (CPSs) in which sensor suites would poll data during the manufacturing process to help verify that the systems within the cell were working correctly. Furthermore, the sensors would form part of the quality chain, demonstrating that the manufacturing process had taken place within the expected margins. The systems would be on-line during the manufacture but would remain running through the transport and eventual operation of the smart frame. This would link the quality and data chains from manufacture to in-service and potentially up to disposal at End of Life (EoL). The sensor suites would be linked in a web of Internet of Things (IoT), and show readouts to the engineers and managers through a dashboard. The work undertaken by the RE focused on this latter part of the project. The frames are considered smart, due to the close integration of sensor technology that could be polled at any point in the S-Frames manufacturing, in-service or decommissioning life-cycles.

Dash-boarding is now a more common practice, and can be seen across business as a method for displaying either real-time or historical data with the aim of aiding the decision making process. One of the central tenants of this is that the more informed you are the better the decision you can make. The project described below focused on providing not just data but also the relevant context. The addition of context will aid in the conversion of the data into more readily understood information. This was achieved through the use of VEs.

The purpose of the REs involvement in the S-Frame project portrayed in the following sections is solely to demonstrate the ability to pull sensor data from existing sensor systems, and display them in a context sensitive manner. In plane English the task was to show the sensor data in close proximity to the digital twin of the project. The overall task of the discussed project was to assist in showing the RR graduates *method of difference* project using VE. The project in no way reflects the manufacturing methods or sequencing of RR and the S-Frame project, and should not be considered an accurate representation of the project in any fashion. As such the choices of components, locations and methods is purely arbitrary. To the best of the REs knowledge Mixed Reality (MR) has not played any other parts within the development of the S-Frame concept.

6.2 Task

The task set by RR was to create a proof of concept model of the S-Frame to improve understanding of the Smart Frame idea and promote it across the company. To achieve this a team of graduates were assigned to create a model of a Smart Frame that would demonstrate some of the advantages it could provide for the company. The team decided to use Lego® as the modelling medium to show the shape and layout of a Smart Frame. A Texas Instruments (TI) SensorTag CC2650STK (TEXAS-INSTRUMENTS, 2018) was chosen as a sensor suite to give: orientation, temperature, humidity, luminosity, and time. A small camera sensor was used for imaging, and used with image matching to demonstrate if components were in the correct position. The processing power would be provided by a Raspberry Pi, and connectivity by a small WiFi box. As previously stated, the RE was brought on board to show how the data from the smart frame could be displayed within a VE that could be viewed using a variety of hardware solutions.

6.3 Objectives

The objectives of the RE were twofold:

1. To display the sensor data created by the smart frame model within the VE
2. To create a VE that represented both the smart frame and the work done by the team of graduates

6.4 Methodology

The TI SensorTag would collect the sensor data and transmit it to the Raspberry PI which would, in turn, use an external WiFi router to send the data to an Amazon Web Service (AWS) table. The Unity3D games engine would then run a set of JavaScript Object Notation (JSON) scripts to poll the web server and request the data at a specified rate. At runtime, the data would be displayed within the Unity3D engines 3D environment. The team of graduates were responsible for the data flow up to the AWS tables. They planned to use the AWS to supply the data to a dashboard display system. The RE was responsible for adding the context to the data with the aim of turning it into information.

As an end product, it was decided that it would be ideal if the application could be published to multiple platforms. This would enable users to both see and interact with the product on a desktop or a tablet. The predominant reasoning behind this was that if the project was displayed to external parties, and they did not have permission to access a RR laptop, they could still view and interact with it without having to download it to their own tablet or phone, which might not have the capability of running the demonstration. So a tablet or phone capable of running the projects VE could be used with the demonstrator.

This led to the choice of using a games engine, which already has the native capabilities to publish to multiple platforms. The chosen engine was Unity3D (TECHNOLOGIES, 2014) as there was no charge for using it for development, and there was an existing AWS Software Development Kit (SDK) for integration.

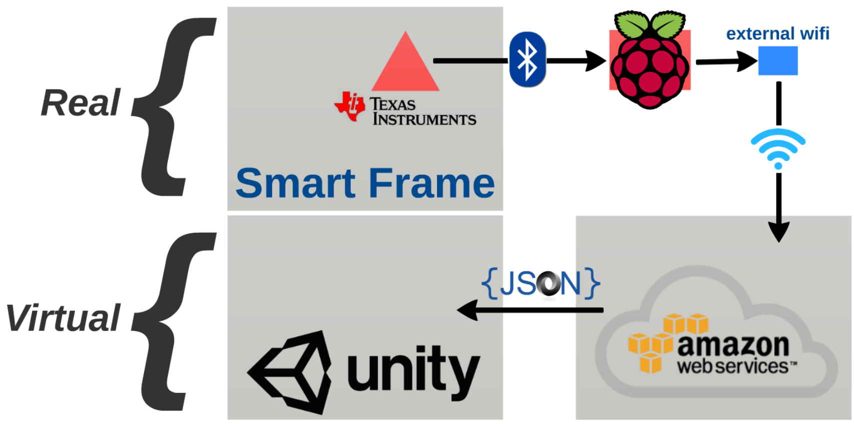


Figure 6.1: Data flow diagram for Smart Frame project

6.4.1 Data Display

The pipeline for the data was made easier with the help of two code bases, the foremost being the AWS SDK which gave access to the tables and information through a variety of function calls. The code base for the SDK is large and complicated, so an additional intermediary package was used. The interpreter package used was Dynamo Database Helper created by the now defunct OuijaPaw Games LLC (OUIJAPAW, 2018). This package consisted of a large name space of functions that simplified the interaction with the AWS SDK.

Within the Unity3D environment only two MonoBehaviour scripts were required. The first was the DBUnityHelper.cs which handled the setup and connection between Unity3D and the AWS

DynamoDatabase. The second was to create sets of First Out (FO) queues which are drained at each render frame. These were the transport mechanisms between the AWS and Unity3D, and kept the polled JSON data messages out of Unity3Ds main render threads, which would have caused significant reduction in performance. The JSON data packets completed the data flow from the sensor in the real environment to the VE objects within Unity3D (Figure 6.1).

6.4.1.1 DBUnityHelper.cs

The DBUnityHelper.cs script is attached to any given Unity3D object and has a set of public variables that allows for credentials, region and other options to be set up within the Unity3D editor itself (Figure 6.2).

```
        public bool showDebugMessages = true;

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ Connection ------")]
#endif
        public ConnectType connectType;
        public DBConnect.region regionEndPoint;
        public string IdentityPoolId = "Use for Cognito ID";
        public string AccessKeyID = "Use for DDBLocal / remote DDB";
        public string SecretKeyID = "Use for non-cognito remote DDB";

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ DynamoDB Local ------")]
#endif
        public int localPort = 8000;

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ Proxy ------")]
#endif
        public bool useProxyForConnection = false ;
        public string proxyHostName = "";
        public int proxyPortNumber = 0;
        public string proxyUserName = "";
        public string proxyPassword = "";
```

Figure 6.2: DBUnityHelper.cs public variables

The script uses the `void Awake` function to initialize a connection between the `gameObject` and the specified DynamoDatabase set within the engine (Figure 6.3). The `void Awake` function is used instead of the `void Start` to allow the VE creator to call the link on object initiation rather than when the VE starts itself. The `void Update` function is then used to enqueue data frames from the AWS database table; the `void Update` function runs each time the engine renders a new set of frames. Adding the data retrieval lines into this function allows the database to be polled as fast as the game engine is running. In the S-Frame context that was unnecessary and a separate variable was created to add a poll interval. The full script can be found in Appendix D.1.

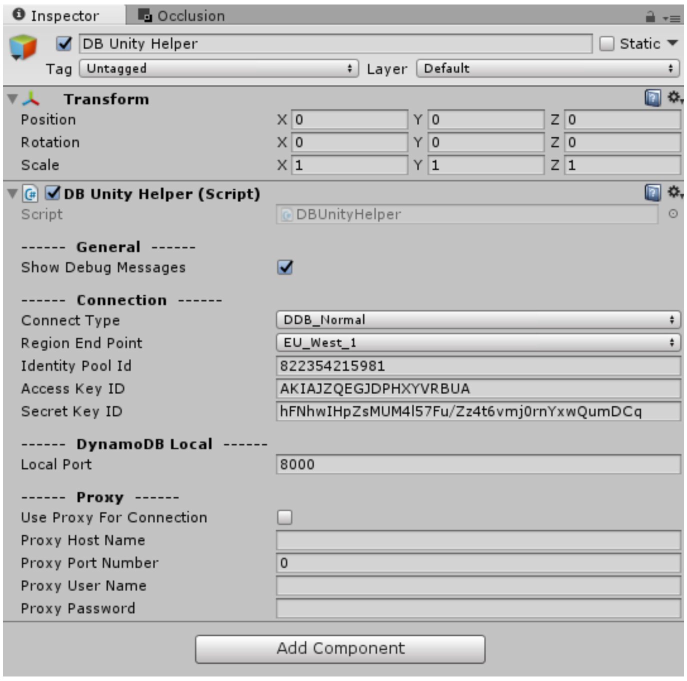


Figure 6.3: DBUnityHelper public variables in the Unity3D engine

6.4.1.2 DBUnitySFrame.cs

The second script for importing the data from the AWS calls directly from the AWS SDK (Figure6.4).

```
using System;
using System.Collections.Generic;
using Amazon.DynamoDBv2;
using Amazon.DynamoDBv2.Model;
using Amazon.Runtime;
using UnityEngine;
```

Figure 6.4: DBUnitySFrame.cs using AWS SDK

Extending the DDBHelper namespace with its own specific class, it first tests that the table exists on the AWS. The void Start function checks that the table of given name AWSTableName exists, which if this returns true, then repeatedly rescans the table at a specified interval. The rescan function calls on the functions within AssetQuery.cs which is where the data is stored in variables within Unity3D. This reduces the load on the processor by only polling the web server at a specified interval. Finally the script has a public void OnClick function that allows for the table scanning function to be called from an object, such as part of a Graphical User Interface (GUI) (Figure 6.5.

```
namespace DDBHelper
{
    public  class  DBUnitySFrame : MonoBehaviour
    {
        public  string  AWSTableName = "unity";
        public  bool  AWSConnection = true;
        public  bool  ScanTable = true ;

        void  ReScanTable()
        {
            DBWorker.Instance.ScanTable(AWSTableName, CapacityType.INDEXES);
        }

        void  Start ()
        {
            if  (AWSConnection)
                DBConnect.TestAWSConnection(AWSTableName);

            if  (ScanTable)
            {
                InvokeRepeating("ReScanTable",2,15);
            }
        }

public  void  OnClick()
        {
            DBWorker.Instance.ScanTable(AWSTableName, CapacityType.INDEXES);
        }

    }
}
```

Figure 6.5: DBUnitySFrame.cs Monobehaviour functions

6.4.1.3 External/InternalSensorTag.cs

The data is finally displayed within the VE using labels set to the 3D objects of the SensorTags within the VE. The TextMesh properties are set as the returning values of the AssetQuerys. This was performed on void Update to ensure that the data was updated in the VE as soon as the database was polled. A more complex arrangement could have been constructed where the values were only assigned on Event such as the data base being polled. However in early testing this method proved sufficient (Figure 6.6).

In the real world model two SensorTags were used, one above and one internal to the frame. This was mirrored within the VE (Figure 6.7).

6.4.2 Virtual Environment

It was decided that the VE was to be as close to the model as possible. This was important so that any users would realise immediately that the VE was a facsimile of the model itself. This will aid

```
ï»¿using UnityEngine;
using System.Collections;

namespace DDBHelper
{
    public class InternalSensorTag : MonoBehaviour
    {
        Transform labelHolder;

        void Awake()
        {
            labelHolder = transform.Find("LabelHolder");
        }

        // Use this for initialization
        void Update()
        {
            var dateTextMesh = labelHolder.Find("Date").GetComponent<TextMesh>();
            dateTextMesh.text = AssetQuery.internalDate;
            var lightTextMesh = labelHolder.Find("Light").GetComponent<TextMesh>();
            lightTextMesh.text = "Light " + AssetQuery.internalLight + "Lux";
            var humidityTextMesh = labelHolder.Find("Humidity").GetComponent<TextMesh>();
            humidityTextMesh.text = "Humidity " + AssetQuery.internalHumidity + "%";
            var tempTextMesh = labelHolder.Find("Temp").GetComponent<TextMesh>();
            tempTextMesh.text = "Temp " + AssetQuery.internalTemp + "C";
        }
    }
}
```

Figure 6.6: ExternalSensorTag.cs assigning the data table values to a TextMesh within the VE

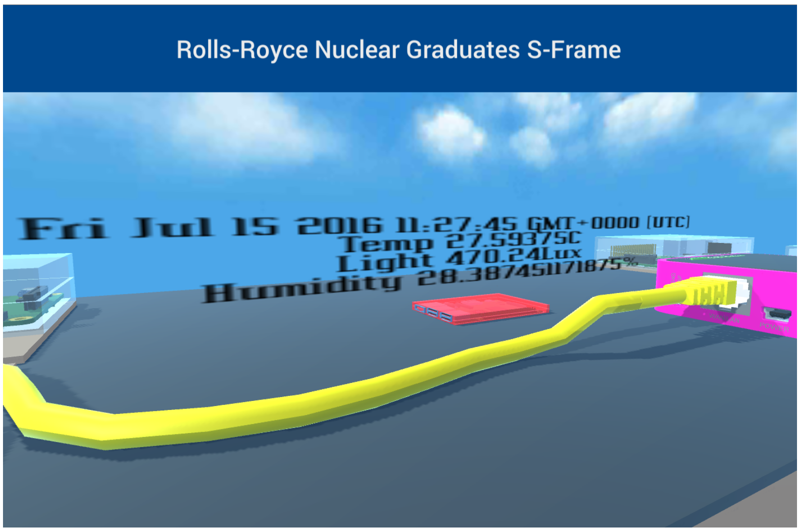


Figure 6.7: The sensor data displayed within the VE

quicker comprehension that the Virtual Reality (VR) model is a digital twin of the real model. With this in mind a one to one Computer Aided Design (CAD) model of S-Frame was created. This digital twin would demonstrate the same principles as the S-Frame test model itself, with the added functionality of being published on the Android, Linux and Windows platforms.

6.4.2.1 Functionality

The demonstrator model was designed to check whether the Lego *nuclear* components were in the correct place, to allow for the same functionality users within the VE, must be able to:

- pick up and move the Lego® *nuclear* components
- locate where the components should be moved to
- move and place the components in the correct location
- verify the parts are in the correct location from the same camera locations as on the Lego® model itself

For the VE to function correctly, additional functionality would need to be created. The engine would need:

- a GUI that was easily and immediately understandable by any user
- a series of reset points for the component parts and the user
- a level of parity with the real frame that users would understand the relation
- the information from the sensor must be visible and legible for the users

6.4.2.2 Virtual Environment Scene

All the models were drawn up in 3dSmax (AutoDesk, 2017a) their polygon count was reduced using mesh optimisers to lessen their impact on the render engine. The models were then imported into a blank scene within Unity3D, the lighting was built and baked and a simple sun cycle animation was implemented to demonstrate dynamic shadows. All non-movable objects were designated static to allow for further in-engine render optimisations. The scene was then "painted" using standard Unity3D materials and imported textures. With the stationary elements completed, a further dynamic element was created by daisy chaining objects together to make flexible cable. With the devices all set up, plugged in and ready to go, the movable assets were imported.

The scene-graph in Unity is an organisational element and method of parenting objects to others. Unlike other graphics engines it does not function as a render time scene-graph. The layout and structure of the scene-graph does not affect the render time or render methods. These are handled separately by the occlusion and lighting methods, which can be baked in once the scene has been created.

The scene-graph is important for organisational reasons, and as such, the use of it in this VE was to separate the parts into environmental meshes, S-Frame meshes, lighting, and movable parts. Having movable parts attached solely to the root of the scene-graph means that they are not a child of any other object and therefore cannot be affected by any transform of their parent object. It also reduces the search time of any Find function used to locate the object or scripts attached to that object.

The Lego® bricks were assembled to roughly represent components that could be made in a nuclear manufacturing plant. These consisted of a large long *"heat exchanger"* (Figure 6.8), two large tower *steam generators*, and a flat *"pressure vessel head"*. Collision meshes were added to all of the Lego® objects, as well as the floor and the steel construction of the S-Frame. Once the objects had collision meshes associated with them, then a dynamic body script was added to create the effect of gravity. Gravity in Unity3D works in a similar fashion to many aspects of the game engine: each object that gravity affects must be given a script to let the engine know how the gravity variable works. The script is the RigidBody which allows for specification of Mass, Drag, Angular Drag, a boolean for Use Gravity, and others including Collision Detection. The latter was set to Discrete, this was to ensure that any physics algorithms that ran off collision data were only called once. With continuous detection the refresh rate of the physics model can cause an issue where the collision meshes become overlapped between frames. This causes the Mesh Colliders of the objects to be entangled, which in turn leads to some very unrealistic results from the physics dynamic calculation for both objects.

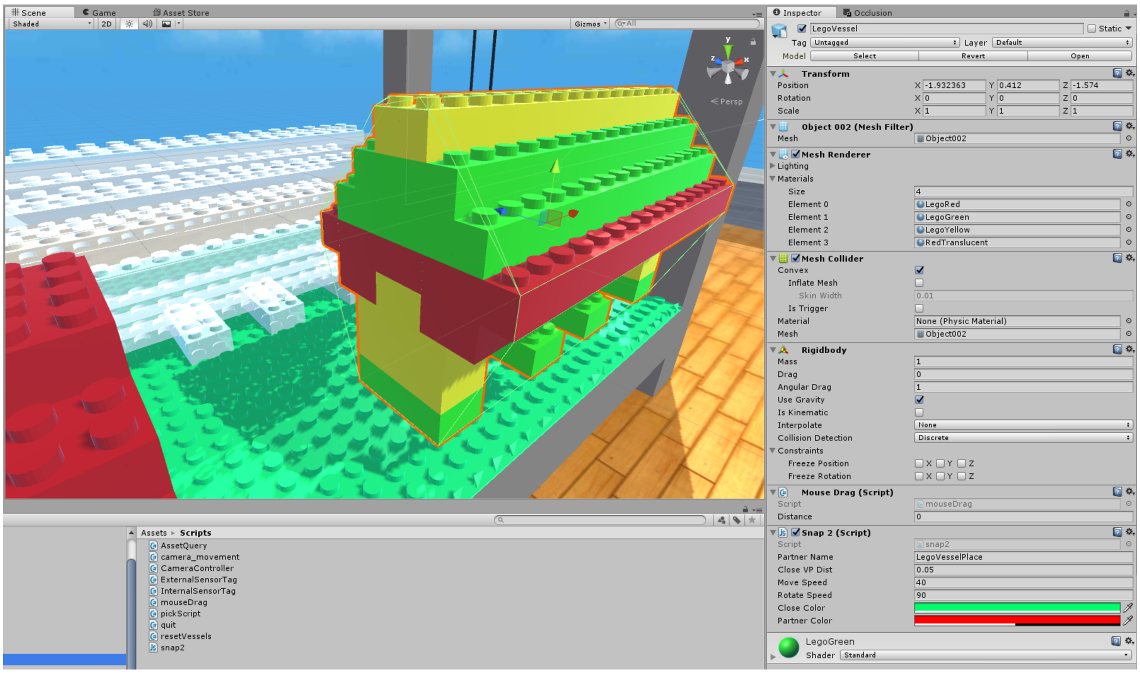


Figure 6.8: "heat exchanger" with simplified Mesh Collider, RigidBody, Materials and other custom scripts

To mirror the functionality of the real world model, transparent versions of the Lego® assemblies were created in the locations they were supposed to be placed in; this would allow any discrepancies in orientation or location to be easy to spot. Camera locations were also set up in the same locations and with the same rotation settings so that the VE images would look similar to, if not identical to the real world models. However, instead of using image processing software to run a *diff* calculation between the two images, the VE shows the locations where the parts should be using translucent materials. The user can then discern the difference by checking from the locked camera positions. The game engine could give an exact difference in the translation and orientation of the free-to-move Lego® Vessel and its translucent counterpart. The translation is given as x,y,z Transform coordinates, and the orientation by w,x,y,z Quaternion. Displaying these numerical results to users was deemed to be unnecessary as the snap2 javascript (discussed later) would display them graphically.

Moving around the scene was accomplished by using a free camera script. The controls were set to a gaming standard mouse look and w,a,s,d. The script can be found in Appendix D.5. To accompany the free moving camera, a second script that would allow for the switching between scene cameras was created. The script used an array to disable all other cameras except the next one in the array. This rotated the active camera view through the array on either keypress or GUI button press. The full script for camera control can be found in Appendix D.6.

The missing part of the puzzle was how the user would interact with the Lego® components and move them into position. The script would need to allow the user to pick up and move the objects on both desktop and tablet/phone platforms. The solution to this was a simple but elegant mouseDrag.cs script (Appendix D.7). The script used two private functions OnMouseDown and OnMouseDrag. These functions are extensions of the unity System.Collections name space, and do exactly what their names suggest. OnMouseDown is called when the user holds the left mouse button down or, provided the scene has the touchInput module added and enabled, holds his finger to the screen. The function creates a vector called heading which is directly from the camera position to the object position; it then assigns the dot product of this vector and user cameras forward position to a float which will be the distance the object appears from the camera when clicked. The OnMouseDrag function then calculates the mouse position vector using the x,y positions of the mouse and the distance floats. The new transform.position or new transform for the picked object is calculated from the main camera position and the mouse position, before being assigned to the object (Figure 6.9).

The mouseDrag.cs script allowed the users to pick up any objects which had the script attached to it. As the script uses Camera.main in its position calculation for the object, the user could move the object around using the camera movement script. The method proved effective at picking up and moving the objects, but being able to place them into the correct position and orientation was next to impossible. When gravity, collisions and dynamics were removed, getting the location and rotation of the objects to match was impossible. A modern software solution to this issue, was to create a form of *snaps*, to give the object the correct position and orientation if the user could move it into a *snap* region.

The snap2.js is a short Java script that uses OnMouseDrag and OnMouseUp functions to change the colour of the part it is assigned to (Figure 6.11). OnMouseDrag also assigns a coloured material to the partner GameObject partnerGO (Figure 6.10). In effect, when a user picks up the object, it changes to a predefined colour, and makes its partner visible by assigning a pre-defined material to it. This

```
void OnMouseDown()
    {
    Vector3 heading = this . transform . position - Camera.main.transform. position ;
    distance = Vector3 . Dot(heading, Camera.main.transform . forward) ;
    }

void OnMouseDrag()
    {
        Vector3 mousePosition = new Vector3( Input . mousePosition . x, Input . mousePosition . y, distance ) ;
        Vector3 objPosition = Camera.main.ScreenToWorldPoint(mousePosition);
        transform . position = objPosition ;
    }
}
```

Figure 6.9: mouseDrag.cs : use of in built unity functions to allow the user to control the position of the vessels within the VE

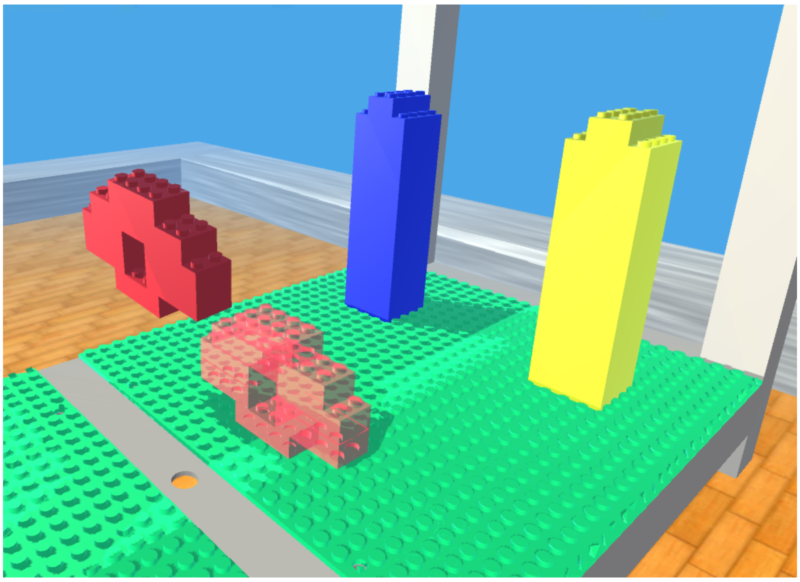


Figure 6.10: Picked object, with its placement location highlighted

```
function OnMouseDrag() {
    var partnerPos = Camera.main.WorldToViewportPoint(partnerGO.transform. position ) ;
    var myPos = Camera.main.WorldToViewportPoint(transform. position ) ;
    dist = Vector2 . Distance ( partnerPos , myPos);
    GetComponent.<Renderer>().material . color = ( dist < closeVPDist) ? closeColor : normalColor;
    partnerGO = GameObject.Find(partnerName);
    partnerGO.GetComponent.<Renderer>().material . color = partnerColor ;
    partnerGO.GetComponent(MeshRenderer).enabled = true;
}
```

Figure 6.11: snap2.js OnMouseDrag function

highlights both the picked object and in this case designates where that object is supposed to be placed. The colour of the object then changes from red to green when the transform . position , an xyz position Vector3, of the object and its partner object are within the stated var closeVPDist distance. This serves to indicate to the user when they can release the object.

```
function OnMouseUp() {
    if ( dist < closeVPDist) {
        transform . parent = partnerGO.transform ;
        GetComponent.<Renderer>().material . color = normalColor;
         InstallPart () ;
        partnerGO = GameObject.Find(partnerName);
        partnerGO.GetComponent(MeshRenderer).enabled = false;
    }
}

function InstallPart () {
    while (transform . localPosition != Vector3 . zero || transform . localRotation != Quaternion . identity ) {
        transform . localPosition = Vector3 . MoveTowards(transform. localPosition , Vector3 . zero , Time.
deltaTime * moveSpeed);
        transform . localRotation = Quaternion . RotateTowards(transform . localRotation , Quaternion . identity ,
Time.deltaTime * rotateSpeed ) ;
        yield ;
    }
}
```

Figure 6.12: snap2.js OnMouseUp function

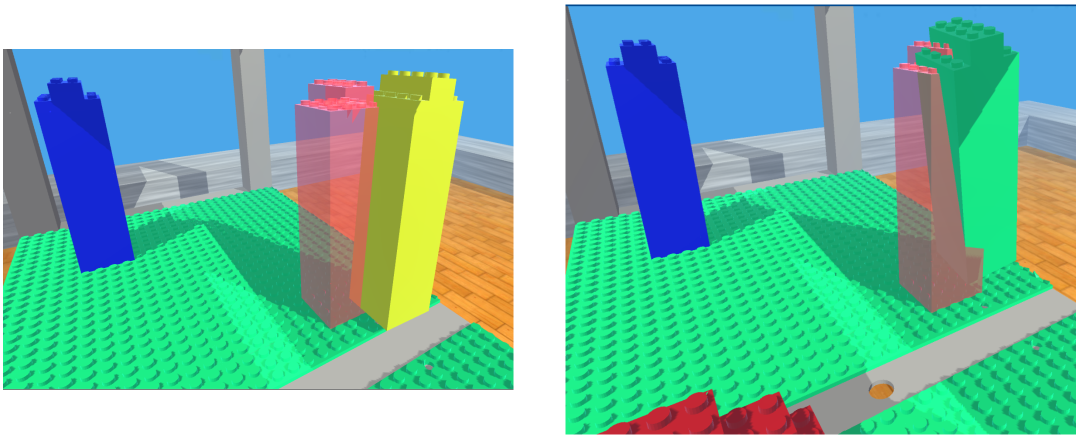


Figure 6.13: Snap2.js script demonstrating proximity of parts and placement

The OnMouseUp function calls the InstallPart function which uses a while loop aligning the transform . position and transform . rotation of the pick object to the one highlighted partner object (Figure 6.12). In effect this means that once the user has moved the object close enough to its intended position, it will turn green and when the user releases the object it will slowly move and rotate to the correct position (Figure 6.13).

This may seem like it is negating the idea that the user should be positioning the object themselves, and observing the differences between where they put the object and where it was supposed to go. The VE is designed as a complimentary piece of software to the real model. As such the real model was where this test of differences was applied, its digital twin the VE was designed to show users where and perhaps how the objects should be installed. In this way the differences between a training and a teaching environment become quite clear.

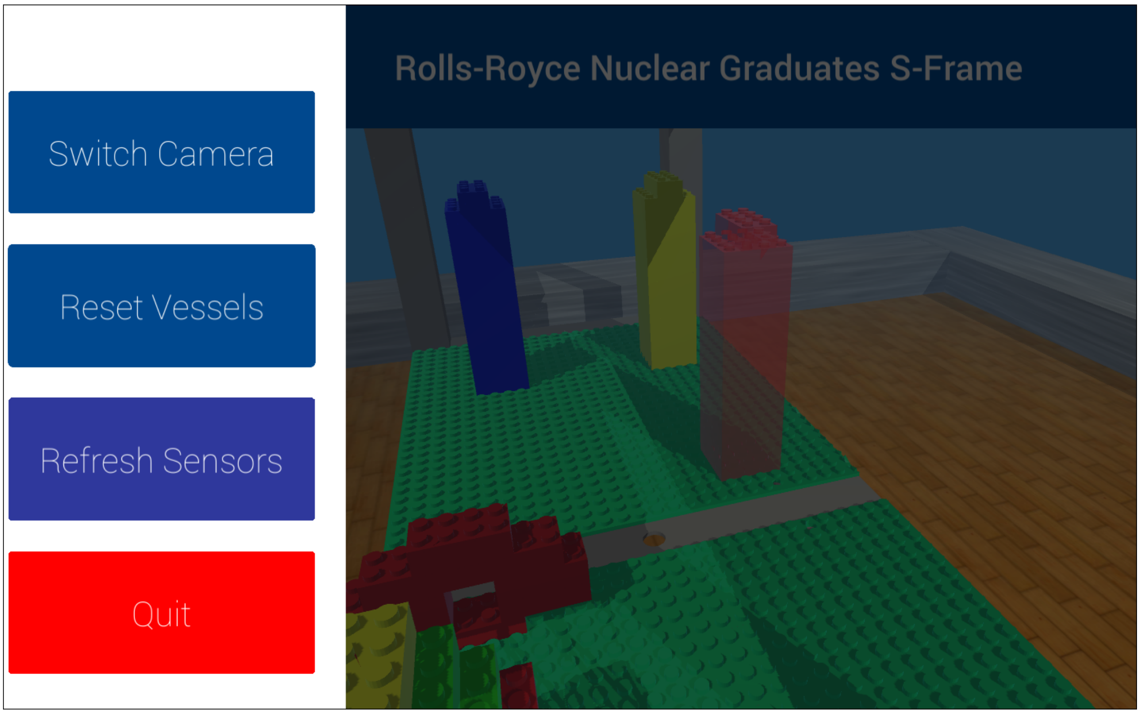


Figure 6.14: GUI for the SFrame application

Using the `snap2.js` scripts functionality the VE now allows for the user to pick, move and place the components in the correct place with ease. For desktop users a set of keyboard commands were set to allow for quick camera position changes, resetting the Lego® vessels and refreshing the sensor information. The key press shortcuts were all set within an `void Update` function, which meant that the engine was constantly checking for the keypress, `if(Input.GetKey("up"))`. This required users to have prior knowledge of what keys to use, and to have access to a keyboard. The GUI system was based of Googles Material Design system, and used a menu button in the top left which would call a frame of buttons to slide out from the left when touched (Figure 6.14). Each button called a separate function, including one that would close the application. This was important when publishing to Android or iOS as, without it, the application would continue to run in the background. As with many applications that use 3D , it was CPU and therefore battery intensive.

6.5 Discussion

Overall the project was well received by Rolls-Royce: every element of the project worked as expected, and it served as an example of how sensor data could be integrated into the S-Frame concept. The VE was an accurate one-to-one representation of its real world counterpart, and carried across the same level of functionality. This made it a true Digital Twin (GRIEVES, 2014), allowing for elements of the unseen real world to be realised in the digital. The project meant that an internet-connected user anywhere in the world could access and assess the data and information provided by the VE. This could allow managers to check in on the status of their manufacturing processes within the S-Frame from one factory, while planning for those outcomes a geographically separate one. The difference between this project and a fictional dashboard, which is solely a database, is the addition of the context to the data. With this, the data becomes readily understandable and more easily digestible because it is surrounded by a representation of its real world counterpart. It could be argued that the closer the verisimilitude of the representation then the easier and faster the decision making process associated may be. Mori et al, first represented this issue in the 1970s paper Bukimi no tani (MORI, 1970). The essence of this argument was we like artificial beings more when they appear more humanlike, but we really dislike them when they appear almost perfectly human with a few subtle flaws. The drop in our acceptance of what looks real is commonly referred to as the uncanny valley. It is this valley that the RE is suggesting should be avoided, and as such the models need only to be representative of their real world counterparts.

In fact, the project itself was criticised by RR for its lack of realism, and a cartoon like aesthetic (Figure 6.15). Which is true, and no doubt compounded by the rotating sun light elements causing moving shadows at regular intervals. This is an element that should have been removed from the final product, as it added little to the simulation, and was first implemented to test its effect on the frame rate on low powered Android devices. The bright colours and lights were mirrored in the digital twins real world version. In this way the project was a more accurate representation of its counterpart than a more *realistic* looking version would have been. However this fails to take into account that the real world model was in itself designed as a representation of the S-Frame Manufacturing Cell. The use of a VE meant that the actual S-Frame Manufacturing cell could have been reproduced. A realistic representation of a manufacturing cell, with an accurate depiction of where and how the sensor data could be useful would have provided a greater sense of context and helped demonstrate what the core of the project was actually about.

A more realistic representation of what the S-Frame Manufacturing Cell could have looked like, would have improved both the reception of the project, and the understanding of what it sought to achieve. It would also have allowed for greater clarity when demonstrating how in-context data can be important. As an example, while welding, the bulk temperature of the metal and surrounding area can be critical to both the metal itself and perhaps also to other components in the area. A solenoid actuator close to the weld joint could suffer from heat damage; a rubber seal may melt, or plastic housing may deform. If the VE could show not only the temperature data but also the temperature data around the pipe itself, then the importance of the spatial context would be easier to grasp. However the discussed issue here is not one associated with correct manufacturing processes, or even how the data capture could be complicated successfully for different welding techniques. Rather it is about what importance the correct display of data in-situ could have.

The manner in which the data was displayed could also have been improved in a number of ways. A simple extension script could change the colour of the temperature text, dependant on the value.

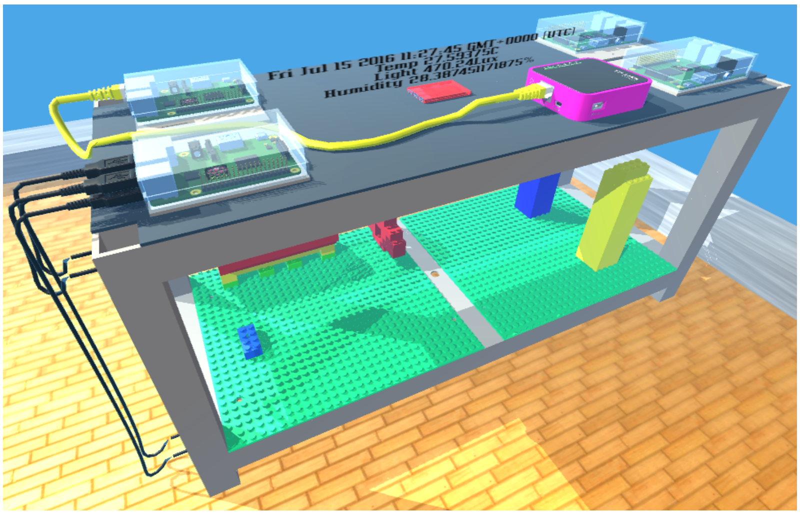


Figure 6.15: The completed VE was criticised for not being life like

Furthermore, it could flash a warning or sound an alarm drawing the user towards the area where the temperature was exceeding a stated boundary value. A further modification could also incorporate the 3D model of a temperature sensor; either a simplified thermometer to make it obvious it is a temperature readout, or perhaps a sliding gauge showing the temperature range which fills as it gets hotter. If a sensor array was being used instead of a single point sensor, then a heat map could be shown as an overlay on the 3D object itself showing both the location and the temperature spread. Rotation of the sensor could be tied to the rotation of the part it is attached to and/or in. Furthermore, animations could be triggered to demonstrate how the part needs to be installed from its current location and rotation.

It is worth mentioning here that the project discussed here was undertaken solely to exemplify the benefits of integrating sensor data sources with a VE. The methods of construction or manufacturing techniques and requirements were not in scope, and as such have only be used as examples. The ability to display data in a context-sensitive manner are demonstrated here, using a method that is low cost, and could be easily transferred to another project. The use of a game engine, such as Unity3D as an endpoint for this data was also demonstrated, and again in a manner that could easily transferred to another project. In the next chapter the Digitally Assisted Assembly Cell (DAAC) further demonstrates that data capture and display in context-sensitive fashion is achievable in a low cost game engine.

As a proof of concept project the S-Frame VE established that the Unity3D engine could provide a simple and practical environment for sharing data across multiple platforms. The community support, the wide availability of custom scripts, materials and even AWS SDK support means that adding additional functionality is easier than it may otherwise be. At the same time it may also create more work as a developer needs to understand multiple code bases to be able to make them function together. Publishing to Windows, macOS and Linux desktop environments, as well as both iOS and Android mobile platforms, is advantageous. Even if configuration is required to adapt the project settings for each platform, the old addage that *results may vary* is certainly true. However the optimisation techniques for the 3D environment, as well as the render engine provided an end-to-end result that demonstrated that live data could be pulled from sensors and effectively displayed within a VE.

CHAPTER 7

Digitally Assisted Assembly Cell (DAAC) - Overview

The chapter introduces the concept of the Digitally Assisted Assembly Cell (DAAC) and how it was created in reaction to the ongoing 4^{th} Industrial Revolution (IR4.0). Preceding from the introduction are a series of sections which give an overview of the methodology used in the creation of the DAAC. The sections cover an initial response to the challenge of tracking users in a room scale environment, before switching to a more robust and working solution. The chapter then discusses the hardware requirements for hosting a Virtual Environment (VE), and aspects of the Human Computer Interaction (HCI). The penultimate section discusses three main areas of feedback between the real and virtual users. Before concluding with an overview of the project as a whole.

7.1 Introduction

The use of Virtual Reality (VR) is well documented in the realm of manufacturing (Section 4.2), where it can provide a functional space for the import of data and information to assist in the decision-making process during the design of manufacturing procedures. In future, it is very likely that the usefulness of Virtual Reality will extend to the collection and display of real-time data from factory floors across the world, to provide in-process monitoring of manufacturing performance. Provided the sensor suites exist to produce the relevant data and information from any pertinent context, Mixed Reality (MR) techniques could be used to display these and interpret the results.

As the IR4.0 spreads further into modern manufacturing techniques, so does the ability to use data to inform decision making, improve best practice and optimise flow around a factory; reacting to issues before they arise and cause problems becomes not just possible but probable. Machines can now not only inform staff when the sensor data deviates from the norm, but can be programmed to react to the change, either adapting its own operational procedures or by broadcasting the issue to other local machines that can then adapt their processes to the change. The network or Cyber Physical System (CPS) creates a Hardware Abstraction Layer (HAL) that is factory wide, and functions as an adaptive mesh for optimising both manufacturing times and cost.

CPS and MR applications now cover a wide variety of procedures within a factory, but one area of data that is currently lacking is information about the factory workers themselves. Systems are being created that aid factory workers, such as helping them with picking orders, and displaying in-situ work instructions through the use of Augmented Reality (AR) and/or projector systems.

Currently the workers' movements around and within a manufacturing cell represent a large gap in the factory data set. The idea of tracking workers is not new: Siemens pioneered the use of Kinect sensors to help optimise nuclear repair workers' movements, in a bid to lower the radiation dose they receive. Other efforts have been made to introduce users to VEs by using motion trackers and markers similar to those used in the motion picture industry. However this information is costly to obtain, and factory conditions are rarely met when the users are being recorded. This can reduce the overall value of the exercise.

Tracking people within a manufacturing environment would help capture the current best working practises. Recordings of best practise could be reviewed and evaluated, and then used to help inform and teach future users. Within the nuclear industry and other manufacturing industries which have an ageing workforce, this could help ameliorate the loss of value from retirement, by capturing expert practitioners' knowledge gleaned from a lifetime of experience in the industry. It was with this idea in mind that the DAAC was first projected.

First thought of in early 2016 by Chris Freeman, Michael Lewis, and the author, the Digitally Assisted Assembly Cell (DAAC) sets out to capture best practise and preserve it in a manner in which can be passed on indefinitely. To do this, it is designed to perform three major functions:

- Capture human activity within a manufacturing cell
- Duplicate the activity within a virtual environment
- Provide feedback mechanisms from the virtual back to the real environment

Human tracking is an established technology, but with a wide range of available solutions.It aims to create something both new and useful for the manufacturing context, as well as a method to feedback directly into the manufacturing cell from the VE. This bidirectional link should serve as a feedback loop between the real manufacturing cell and its virtual digital twin.

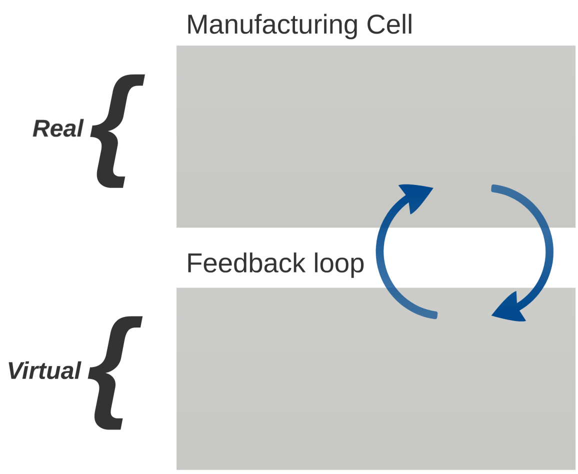


Figure 7.1: Digitally Assisted Assembly (DAA) Cell Concept

In the S-Frame project, data was transmitted from the real into the virtual. The DAAC would extend this even further by transmitting data created within the VE back out to the real (Figure 7.1).

The bidirectional link between the real and digital twins would allow for remote collaboration of users within the VE and the users within the manufacturing cell. In this way, not only could best practice be transferred from the users in the cell, but users with access to all the available data sets could help inform the manufacturing cell users, and could take place in the form of work instructions or on the spot training. Remote assistance is a developing field predominantly for in-service or repair work: it is usually restricted to either single or multiple RGB cameras feeding information to flat desktop screens, where a remote user will provide Voice over Internet Protocol (VoIP) instructions to the in-service or repair worker. The DAAC would extend this by having remote users in a full VE with access to a wide variety of data sets pulled from the manufacturing cell. The feedback mechanism would be extended from a single VoIP option to an array of feedback methodologies.

7.2 Methodology

The DAAC can be represented as four main elements, as in Figure 7.2.

- Manufacturing Cell
- Virtual Environment
- Human Interaction
- Feedback loop

To create a digital twin of the manufacturing cell and its users, a method for tracking the users was required. Real time tracking data would then need to be imported and represented as an avatar into a VE that contained Computer Aided Design (CAD) and other representations of the non-human cell components.

The human interaction element would need to include both the users within the manufacturing cell and the users within the VE. The VE itself would serve as an intermediary between the two user groups, which would send data through a feedback loop, running one way by human tracking and another by projection of data sets and VoIP. Ultimately it would be elements of the VE that would be re-projected back into the real.

Work on the S-Frame had provided enough of a background in creating virtual environments that it was decided that initial research and development should focus on the human tracking element of the project, to discern what capabilities already existed. For a more detailed description of the final solution on used within the DAAC please refer to section 8.2 of Chapter 8, or for more general background research regarding the topic of tracking see Appendix B.

7.2.1 Tracking

Human tracking can now be done in a wide variety ways, ranging from simple geolocation by Global Positioning Satellite (GPS) or other sensors such as known WiFi and cell tower locations, to full body motion tracking. An in-depth look at different sensor types and tracking methods is

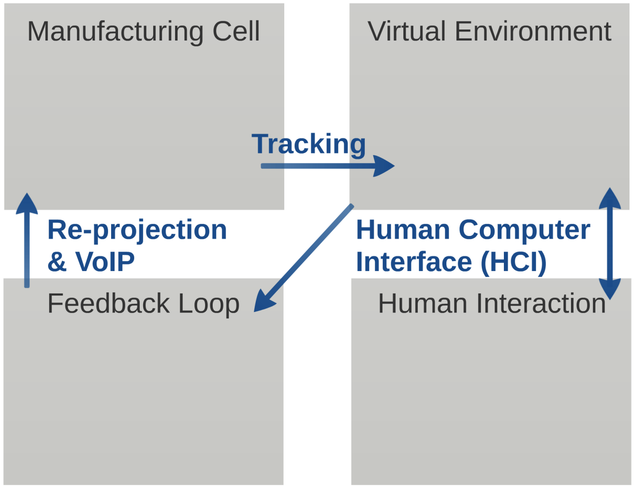


Figure 7.2: Four main constituent parts of the DAAC

covered in Appendix B.

At the cross roads of cost and tracking solutions sits the Microsoft Kinect v2 sensors. These sensors are particularly attractive for use within manufacturing cells as they do not require any fiducial markers or other sensors to be worn by the tracked user. This allows them to move unencumbered within the manufacturing cell, and does not require preparation or prior knowledge before a user can be introduced to the manufacturing cell. The sensors also have a large group of enthusiast developers writing software and programs for the Kinect sensors. This provides an large knowledge base that can be drawn on while developing new applications for the sensors.

Microsoft has ceased to support and develop for the Kinect systems, so although the existing Software Development Kit (SDK) and hardware will still work well together, any improvements will have to come from another source. Its possible that the Free and Open Source Software (FOSS) community will come together and write post process wrappers for the existing SDK but it is more likely that future work will encompass existing FOSS alternatives like libfreenect (**librefreenect**). Which are discussed in the next section.

7.2.1.1 Kinect V2 and the Robot Operating System (ROS)

The Kinect sensors are a conglomeration of different sensors that are coordinated and provide the end application with the required data sets (Figure 7.3). The sensor data is controlled by the Kinect Drivers, and then combined and interpreted by the Kinect services layer. In turn, this provides the Kinect SDK with the combined data sets, such as skeleton data, that can then be harnessed by the application (Figure 7.4).

One of the limitations of the Kinect v2 sensors is that a single computer can only host a single sensor, severely limiting the area that can be tracked to a single sensors capture area. If a user is to be tracked across an entire manufacturing cell, multiple sensors would be required, and

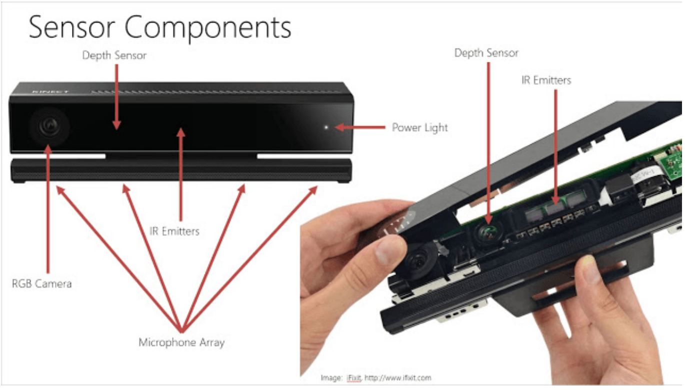


Figure 7.3: Sensor Components of the Kinect v2 (IFIXIT, 2018)

consequently, multiple computers each running one sensor.

A Microsoft product using multiple Kinects to track a large area would require multiple Windows licenses and hardware able to run at a correspondingly high enough specification. For something which would only be a sensor controller, this at first seemed expensive and somewhat excessive. Fortunately, the Open Source community presented an alternative in the form of an open Kinect project that ran on Linux (H. MARTIN, 2018). The libfreenect Linux project mirrored the abstraction layers of the Windows software suite up to the driver level (JOSH BLAKE, 2018), which would provide the requisite data streams, but the service layer and the SDK layer would still need to be developed.

Using Linux means a reduction in cost, as most Linux distribution is FOSS, and can be run with much smaller processor overheads. Since libfreenect is open source itself, then the data stream could be culled at the driver level, which would free up even more processing resource, potentially reducing the hardware requirements even further.

There remains an ever prevalent issue within the nuclear industry about using open source code within the nuclear industry (VENTÄ et al., 2007). However even though the manufacturing processes are strictly controlled by regulation and must conform to a wide range of codes and standards, there is little to suggest that Open Source software could not be used as part of the technological solutions for manufacturing. In fact many CPS and other systems are already running variations of Linux and other open source technologies.

That is not to say that providing the source code for the programs you are running would be entirely safe or recommended if these systems had access to any exposed networks. The ability to see the source code of these systems could allow external operators unique and precise attack vectors into the systems. This is normally mitigated within the open source community by having a critical mass of users and maintainers that update security issues as and when they are identified. This may not be possible for systems that are bespoke or have a small user base since the number of people reviewing and updating the code would be too low to ensure a satisfactory level of security.

The use of open source software in areas where the code would be widely used and supported would allow for anyone to update the software from anywhere and would also hugely increase the likelihood of identify and fixing vulnerabilities not just for your own company but for all companies that use that software. Clearly the selection of what systems are suitable for low cost and open source solutions will be difficult, and only made harder by ensuring that the proprietary systems a company use can communicate with the open source ones. These are issues for networking and computer system specialists and at this level outside the remit of this thesis.

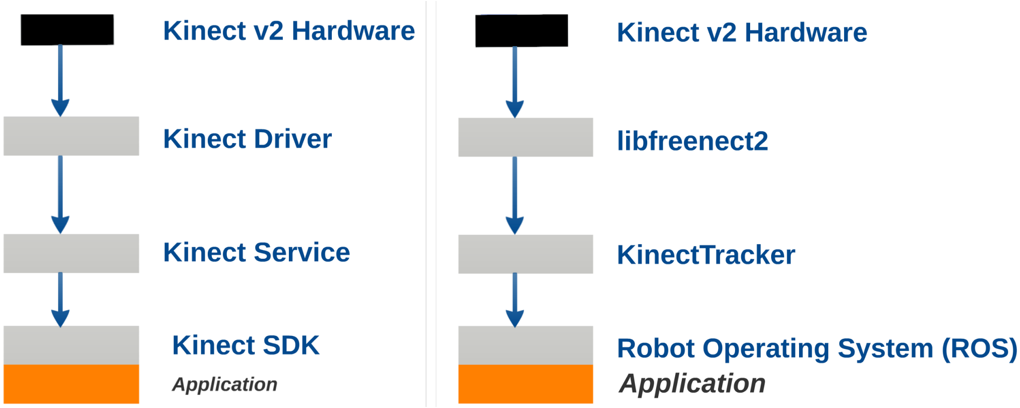


Figure 7.4: Kinect v2 abstraction layers on Windows and Linux

A low powered brix computer (GIGABYTE, 2018) running ubuntu was connected to a Kinect sensor. This provided the raw data which needs to be interpreted into a skeletal data set and could then be combined with other Kinect data sensors before being imported into the VE. To achieve this a Robot Operating System (ROS) wrapper program that harnessed the OpenNI NiTE 2.2 libraries was used to convert the sensor data into joint sets with Cartesian position vectors and rotation quaternions. The joint sets were then broadcast across a ROS network. The KinectTracker program (STEPHEN REDDISH, 2018) was capable of tracking multiple people, and transmitting their skeletal data as individual frames out across the ROS network.

A single Kinect can only track over a limited area and from a single direction, which means a lot of human movement is occluded by the body itself. To track a larger area multiple Kinects and brix computers were used across a network (Figure 7.5). The ROS was used to broadcast each frame of skeletal data from each brix on the network. Each frame was then imported into a global space hosted on one of the brix themselves. As each sensor was positioned independently of each other to maximise coverage it meant that the global space would have as many skeleton data sets as there were Kinect sensors.

The solution to this problem was attempted by creating areas of overlap for the kinect sensors themselves, which would mean they would be recording the skeleton data at the same time and location. The difference in the skeletal position and rotation could then be used to find the vector differences in the sensors location to each other. When taken across the network this meant the sensors relative positions and rotations to each other could be calculated. This would also remove the common issue of line of sight trackers that is encountered through object occlusion, preventing a sensor from recording a users location. Multiple sensors from multiple angles should ensure that enough data is collected on the user to ensure that their location or location of their joints is known. This only becomes unreliable if the tracked individual becomes occluded from all angle, which could happen if the user climbed within another object such as a box or tube. At this point the tracking systems would fail completely, and the user would be untracked.

The brix that was combining the skeletons sets into a global tracked space, could then run those position and rotation vectors as transforms on the data sets from those sensors, and a global space with multiple skeletons aligned on top of each other would be accomplished. The final step of taking the average result of all the transformed data sets for each frame would render a single skeleton data set in a global frame of reference.

Unfortunately despite the Research Engineers (REs) best efforts the final rotation quaternion calculations proved more than the hardware could complete in real time, and the optimisation techniques beyond his abilities. A global space with multiple aligned skeleton data sets was obtained and was readily viewable across the ROS network using the programme reviewer, but without the homogenisation into single user skeleton data sets it would prove impossible to import into a VE.

The RE decided to use Quaternions as a method of storing the rotational joint data as it is industry standard to do so within the world of computer game development. The reason for this is that cartesian coordinate methods suffer from something known as gimble lock. This is where, through rotation of the object, two axis become aligned and as such a rotation in one angle is now the same as in another. Quaternions avoid this issue, but using a fourth angle, which is in the imaginary rotation plane. The imaginary quaternion plane, can be thought of in a similar fashion to that of the imaginary plane in complex numbers. The fourth imaginary rotation is often annotated as *w* i n a coordinate set of *i,j,k,w* and is used to prevent the possibility of gimble lock.

Despite its early promise of network utility and lightweight platform, the problem of integrating multiple Kinect 2 sensors into a single global space proved insurmountable. Fortunately a group at the university of Ulm were developing a similar tool on the windows platform (RIETZLER et al., 2016). The team were kind enough to grant the RE permission to use the early development binaries for the DAAC project. The system worked in a very similar fashion to the ROS networked solution but ran on the Windows platform (Figure 7.5).

7.2.2 Other Cell data

The manufacturing cell was set up to replicate an area for sub-assembly on a wing skin. This involved a large frame that held the wing skin in place, a variety of smaller tool trolleys, and the DAAC cell frame and equipment itself.

As with the S-Frame project, a variety of other sensors could be put in the cell so that their data

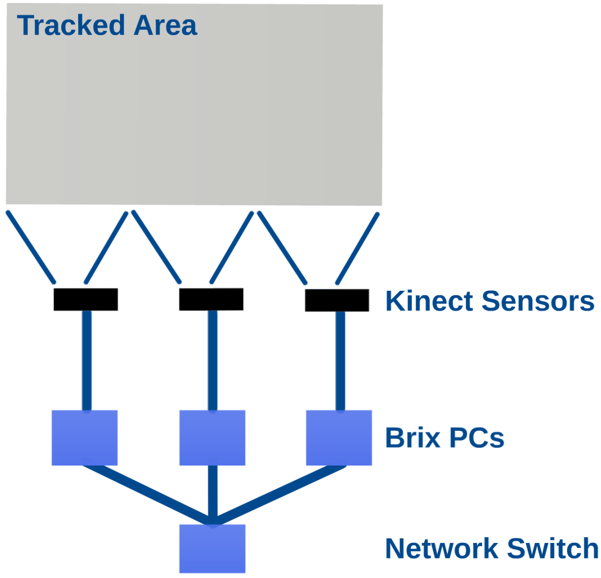


Figure 7.5: Hardware configuration for tracking users in a wider area

could be imported into the VE at a later date. With each Kinect sensor being the host of multiple sensors already, these were used instead to import a real time RGB feed from their on board cameras, along with real time 3D mesh scan data. Network bandwidth limitations meant that only a single Kinect sensor mounted on the server was ever used for this.

The Kinect sensors also have microphones mounted on them, and could have been used as part of a VoIP system, however that would have put further strain on the cells network and required some very complicated noise cancellation algorithms to create a clear voice channel. Instead a handheld VoIP device was used, in the form of an Android enabled device, which would connect to the cells network and be integrated into the VE itself.

Finally an array of projectors was inserted in the manufacturing cell mounted on a large metal framework which would hold the other hardware elements which would make up DAAC. The array of projectors would be used to project data and information from the VE back into the real manufacturing cell (Figure 7.6).

7.2.3 Virtual Environment

The VE was created using the Unity3D engine, as it was low cost, well supported with a large community and within the REs knowledge base. The engine supports direct import of .fbx files which allowed for a quick pipe line for the import of CAD and other 3D models. The instance of Unity that hosted the VE was on a high specification desktop server mounted into the DAACs frame.

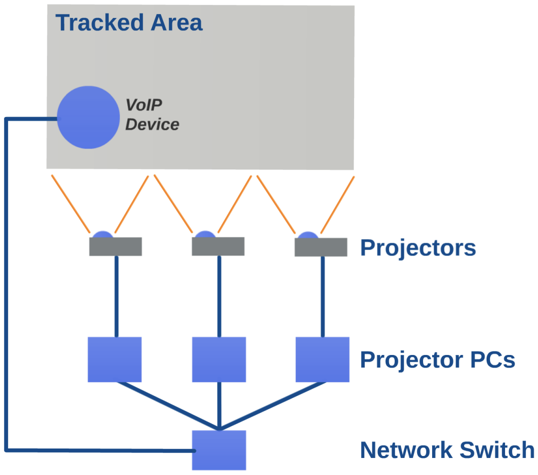


Figure 7.6: Hardware configuration with addition of VoIP device

The server was linked to a network switch, so that lines of communication with the other elements of hardware within the cell would be possible. This Master PC would handle the network traffic from any and all devices connected to the DAAC network. A way to distribute the load across the network was essential, as the VE would need the majority of the computers resources to allow it to run smoothly.

Although the VE was hosted on the Master PC acting as the server, the VEs that users experienced would be hosted on their own devices. This would dramatically reduce the resource load on the Master PC and allow for a smoother and high frame rate experience for the end users. In this way the VE was network enabled so that multiple users could exist within and interact each other and with the VE itself. This allowed for users to connect and interact with the VE by using any Unity capable Android device, as well as using PCs and Head Mounted Displays (HMDs) to exist within the VE. These users could be connected by wired connections to the DAAC network or by joining the network switches WiFi network. If the network had a gateway to the internet it would be possible for users to join from anywhere in the world.

Populating the VE solely with user driven avatars would lack the requisite levels of context to add much value to users. So a variety of methods were used to import data into the VE itself. The most complex method was that of taking 3D laser scan point clouds of the wing skin and converting them into mesh sets that could be rendered at low enough processing cost to make them feasible for use within a VE. The remaining items were created in CAD packages, optimised for the Unity Engine and imported to the VE. A fuller representation of the hardware layout can be seen in Figure 7.7, including the imported data sets.

As previously mentioned in section 7.2.2 the Kinects themselves were used to represent other data sets that could be used within the manufacturing cell. In particular a live RGB feed and real time point cloud mesh data taken from the camera and depth sensor respectively.

7.2.4 Human Interaction

The DAAC incorporates multiple methods for HCI, to accommodate as wide a user base as possible. The Unity engine allows for the publishing of its applications to multiple platforms including Linux, Windows, macOS, Android, and iOS. The allowed for the VE to be published to any device capable of supporting the processing demands of the VE itself. The DAAC demonstrator itself was published to both Android and Windows. Other systems could have used instead of the Unity3D, such as the Unreal Engine 4, however learning a new game engine would have taken a significant amount of time. Furthermore the Unity3D engine proved to be fit for purpose with creating a proof of concept.

The Android application was hosted on a Sony Xperia Z4 tablet, which was used by people within the manufacturing cell to both see into and interact with the users within the VE. The touch based screen allowed for users to look around the environment, interact with specified objects. A slide-out GUI was provided to allow for control over VoIP and to cycle through the work instructions that were projected into the manufacturing cell.

The Windows application was used to host the VE server on the master PC, other desktop users, and any users who wished to use SteamVR enabled PCs with the HTC VIVE Virtual Reality (VR) enabled headset. The interaction for desktop users was based upon established w-a-s-d controls, a variety of keyboard shortcuts along with the ability to use a slide out menu system similar to that available on Android devices. The control systems for VR users worked by using the VIVEs handsets and wide area tracking system. The users could both walk and teleport around the VE using the trigger buttons, whereas the radial track pads allowed for the enabling and disabling of the VoIP systems, and for cycling through work instructions.

Using the above methodologies the ability to affect any and all aspects of interact-able objects within the VE was provided to all users (Figure 7.8).

7.2.5 The Feedback Loop

Communication between the different sets of users is paramount for collaboration, to achieve this three main areas of feedback were created within the DAAC.

The most apparent was the VoIP system which on key press could be enabled or disabled. Earlier iterations of the DAAC used voice activation for the microphones, however in a busy factory environment this meant a lot of noise was broadcast. Furthermore the fact the broadcast channel was always locked open, led to a very large amount of packet traffic across the network, which caused a decay in the synchronisation between the different devices. Another option explored was push to talk, however it was deemed cumbersome for the manufacturing cell users, who would need both hands free to complete tasks within the cell.

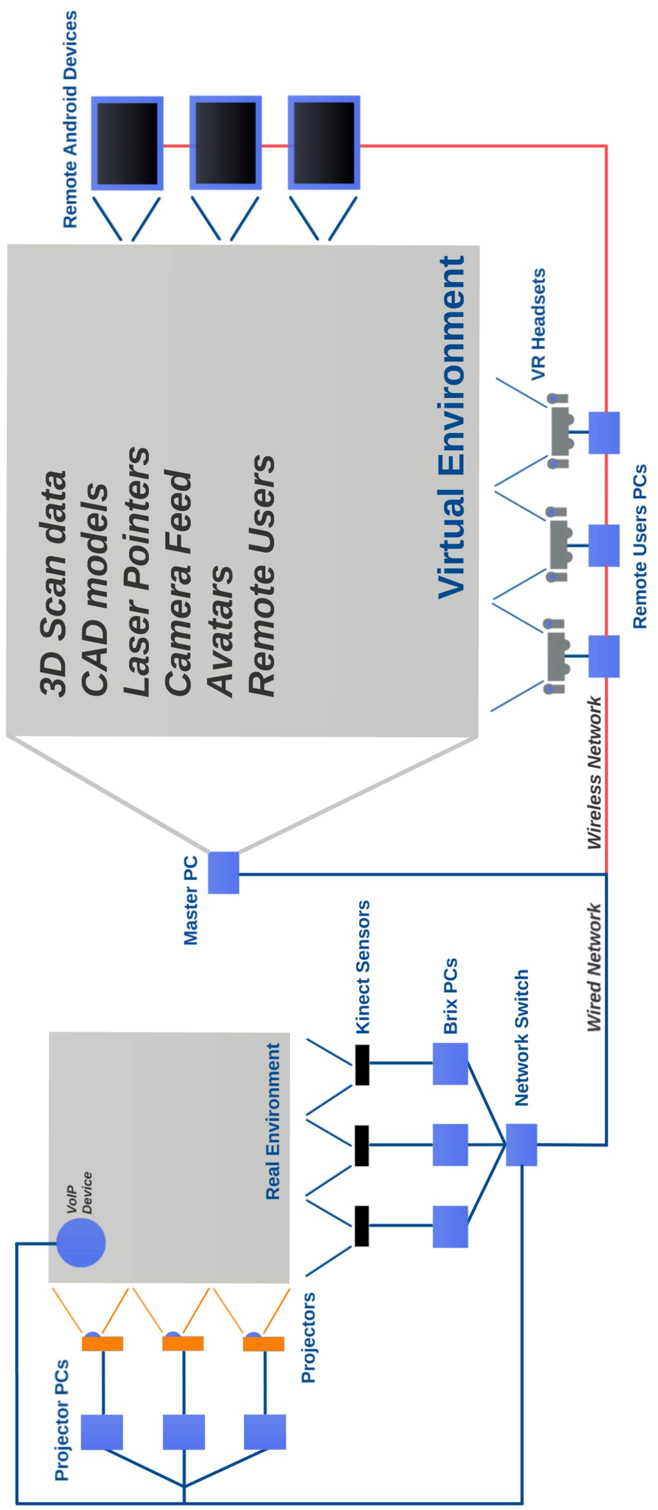


Figure 7.7: Hardware configuration showing items contained within the VE

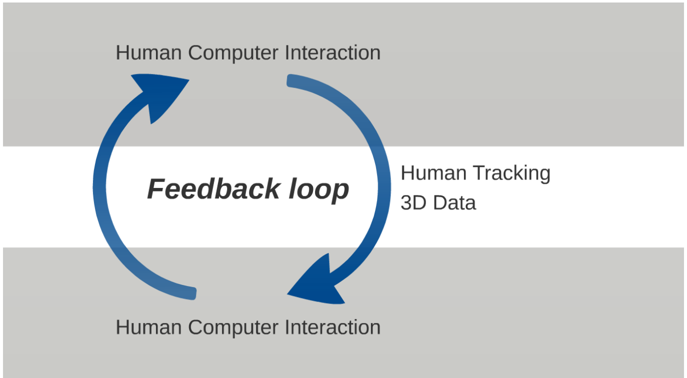


Figure 7.8: The DAAC accommodated HCI from users in both the real manufacturing cell, and the VE

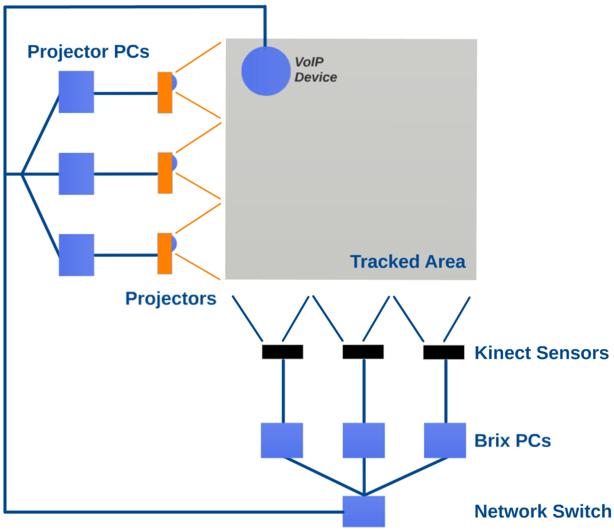


Figure 7.9: Hardware configuration for feedback back into the manufacturing cell

Delivering work instructions to the users within the manufacturing cell could be done using a combination of VoIP and an Android device that allowed the users to see into the VE that hosted the instructions. Although the work instructions could be cycled through and changed at will by the users, they did not provide much more context than traditional paper based work instruction would. To add this context, and allow for in-situ display of the work instructions, a set of projectors were mounted onto the DAAC frame (Figure 7.9). This would project out different elements of the work instructions onto the real world objects they were associated with. As an example a series of cable routes and magnetised boxes were used to demonstrate a set of assembly instructions on the wing-skin. To achieve this the projectors were set up as the default display mechanism for networked users who would join the virtual world. The networked users roles could be multi-functionary, primarily they would be there to assist the workers within the manufacturing cell, using both the re-projection and the VoIP systems. They could also use the VE to collaborate and discuss manufacturing processes and cell layout variations.

The projectors location was mirrored within the VE and set as a camera location. So that when a user jumped to the projectors location within the virtual world they would see same projection area that was covered by the real projector. A culling mask was then applied to the projector camera users so that only the elements that were required to be projected out were visible. In this way a level of projection mapping to the real world objects was achieved.

7.2.6 Overview

The DAAC is a multifaceted project that creates a digital twin of a manufacturing cell. The digital twin is a complete replication of the real, using both CAD and laser scanned point cloud data to create an accurate portrayal of both layout and objects (Figure 7.7). The cell imports the movements of the real manufacturing cell users into the VE using an array of Kinect sensors, while capturing real time RGB, VoIP and 3D mesh data from within the cell. The VE is accessible to external and internal users through the use of Android enabled devices, desktop PCs, and VR enabled systems. A VoIP system and an array of projectors are used to project elements of the VE back into the real manufacturing cell to assist in the delivery of work instructions.

The Real manufacturing cell consists of a large frame that hosts the network enabled hardware (Figure 7.9) and aids in the alignment of the projector systems. It is mounted on wheels so it can be moved around the shop floor. The frame has multiple mounting points, which allows for the projectors and sensors to be held in place at different heights and anywhere along the length of the frame. The frame also hosts a major electrical socket, a small table and a platform for command and control of the system through the master PC.

The main work surface within the manufacturing cell, was a Boeing wing skin, held and bracketed so that the internal wing stringers where visible to the users. A set of coloured rj45 network cables were used to simulate wire looming, and a set of cardboard boxes mounted with magnets, and pictures hooks were used to simulate basic assembly. The work instructions related to the creation and installation of both the wire looming and the cardboard box assembly. The instructions were set out so to maximise the users movement through the cell, to check capabilities of both the tracking and the projection of the work instructions.

The Virtual elements of the projected consisted of a digital twin of all the items within the real manufacturing cell, plus a set of laser pointers so that users in the virtual could indicate areas of interest in the real when they were projected back into the manufacturing cell. The VE also contained several portable screens that could be dragged around. These would display the RGB feed from a single Kinect sensor, the displays from each projector, and a culled version of the displays of each projector. This allowed for VE users to understand exactly what information was being projected back into the real manufacturing cell.

The ability to control the elements within the VE in real time allowed for in-direct but synchronous real time collaboration across the digital divide. Furthermore the ability to record the users action in the real allowed for a-synchronous review of the manufacturing processes within the VE.
The Network of the DAAC formed the back bone of the project (Figure 7.10). With many disparate data sources, transmission techniques, and high processing tasks, it was vital that only the required data was sent across the network to the variety of different users. This involved culling data sets as soon as possible to avoid network traffic clutter, and reduce the demands on the servers CPUs. It also required that the VE server only transmitted information that was required by clients. This reduced the likelihood that the data integrity would break down and the server and clients would fall out of sync. The network was built in a way that would allow it to bridge to the internet, thus allowing users to use the VE from anywhere in the world.

The implementation of all these areas is discussed in greater detail in the next chapter.

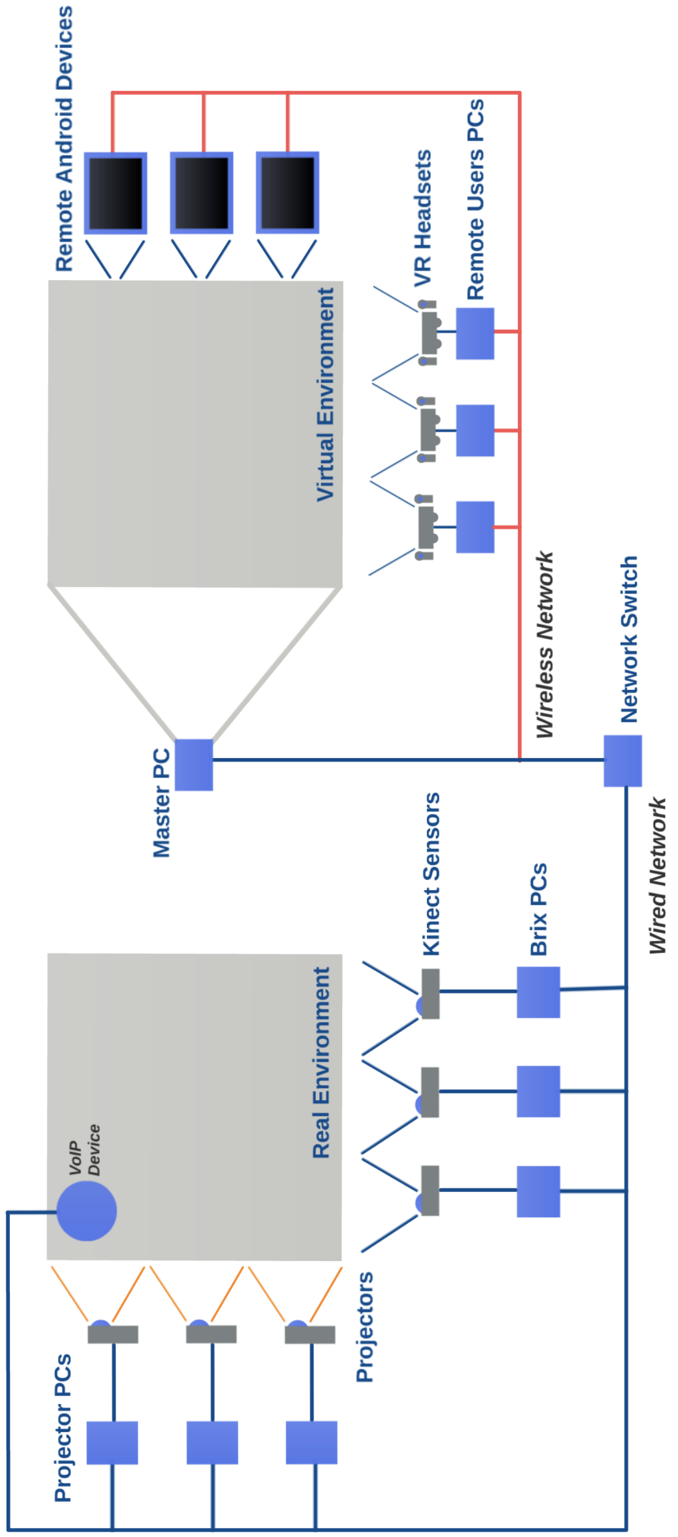


Figure 7.10: DAA Cell hardware configuration

CHAPTER 8

Digitally Assisted Assembly Cell (DAAC) - Implementation

This chapter takes a more in depth look at both the functionality and corresponding methodology behind the Digitally Assisted Assembly Cell (DAAC) project, starting with the back bone of the project: the network communication layer. The chapter goes into greater detail about the Ulm Fusion tracking software used and its abstraction layers. A brief section covers the import of an RGB camera feed, and depth sensor data which is displayed in real time as mesh converted from the point cloud data within the Virtual Environment (VE), from an additional Kinect sensor attached to the DAAC server. Before the main body of the chapter covers the use of Computer Aided Design (CAD) and laser scan point cloud data as data import sources for the VE allowing an accurate Digital Twin to be created. The import of users as data, and avatars as their digital twin is discussed, before a detailed breakdown of the code required for effective Human Computer Interactions (HCIs) across multiple platforms is addressed. Feedback mechanisms and their respective in VE controls are covered such as projected work instructions, and Voice over Internet Protocol (VoIP) implementations. The underlying methodology of how the VE objects, cameras, projectors, tablets, laser pointers, and work instructions were created and implemented are discussed before the conclusion of the chapter.

8.1 Network

As previously stated, the network configuration within the project served as both the backbone and the sole pipeline for transferring data and information between different devices and users. Bandwidth management became important early on in the development of the project. To help handle this, the network was split into protocol and software sources. The tracking software used UDP and REST protocols and one of the Kinect brix PCs was used as a server for the tracking system (Figure 8.1). This reduced the network traffic to the server, as well as reducing the processing load caused by combining the different skeletal data sets. The tracking server then used the TCP/IP protocol to communicate with the VE server and Unity3D. The separate protocols were chosen according to their unique properties. Whereas the skeletal data combination could easily allow for dropped packets and opening and closing connections. Combining the data from multiple Kinect sensors meant that multiple data points existed for the same skeletal joints. The occasional loss of some of this data would not result in the sum of the data points changes drastically or even in practise noticeably. For the link to the server itself, it was useful to know if the connection had

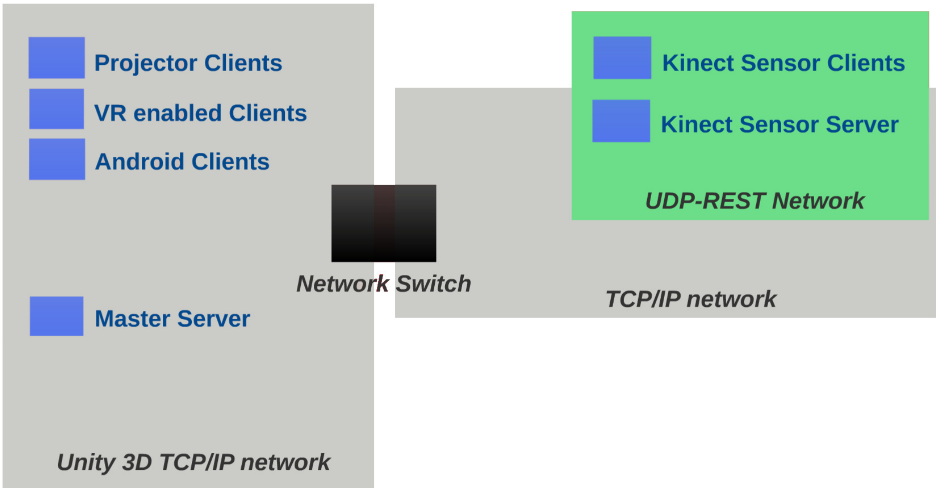


Figure 8.1: Network Abstraction Layers

been closed or not. With these in mind, the properties of UDP and TCP suited the requirements. The other clients and the server itself were connected using the TCP/IP protocol, due to the fact that it was important to know as soon as possible if the connection to one of the clients had been closed.

The Unity3D network engine relies on high quality connections, as it synchronises different data sets from each client and server. The dropping of packets and/or other data sets can lead to the loss of this synchronisation, which in turn would lead to components jumping or teleporting around the VE. This would significantly reduce the feeling of presence or level of immersion felt by the VR users. It would also potentially compromise the delivery of work instructions to the users within the manufacturing cell. One can imagine a circumstance where a user in the VE demonstrates where to drill a hole, or initiates a cut by using the laser pointers; if the work instructions were delayed or misplaced due to back synchronisation, it could lead to the manufacturing users making mistakes.

The network was managed using RDP connections and the NoMachine software to log into each machine individually. This allowed the software to be set up and calibrated using the native Graphical User Interface (GUI). Furthermore, any issues with sensors or projector alignment could be quickly diagnosed and resolved without the use of a command line interface (CLI). Running the clients through NoMachine (NoMachine, 2018) added a layer of network abstraction to the network but the additional load only occurred while the server was connected and logged into the clients by RDP. This meant that the network remained free of RDP traffic outside the set-up and issue diagnosis process.

The network traffic was tightly controlled by ensuring that only the information required by the next part of the network was transmitted. This was done by isolating the Kinect Sensor clients and server to a UDP REST network 8.2. Even within this network the information travelling upon it

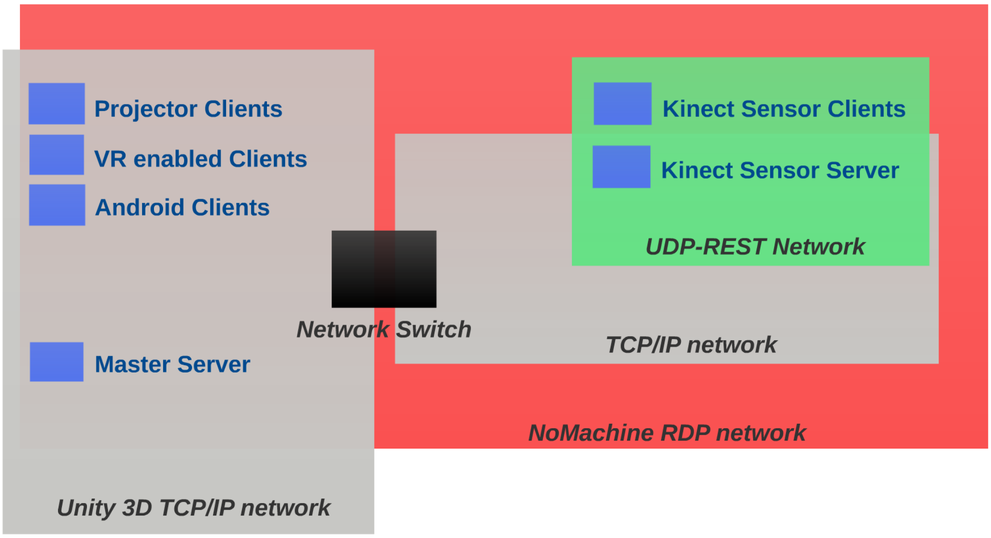


Figure 8.2: Network Abstraction Layers with additional Remote Desktop Protocol (RDP) layer

was just the REST frames of the skeleton joint location and orientation. All the depth sensor, RGB and other data from the Kinects had been stripped out on the Kinect Sensor Clients.

The TCP/IP network was then used to transmit the combined skeletal data of the entire UDP-REST Network. This meant that only one set of skeletal joint position and orientation data was sent across the more exacting network protocol. The multitude of other skeletal data sets having been dropped. This master skeletal data was then imported into the VE within Unity3D.

How the data network and synchronisation of the VE data is further discussed in Section 8.4. It was important to strip out as much data as possible on the network to allow the greatest space for the more important synchronism between different users of the VE.

In order to accommodate a real time point cloud mesh within the VE an additional Kinect was added to the configuration in Figure 8.2. This caused high network load by sending real-time RGB frames and point cloud information across the network. To reduce network traffic a design compromise of running a Kinect service on the server was made. A single Kinect connected to the server created the only real time RGB and point cloud feed for import directly into the VE. This significantly reduced the traffic on the network, but conversely increased the processing load on the server itself. To remedy this, a switch was added into the VE that would enable and disable this feature. The switch worked by not only disabling the feature within the VE, but also disabling the Kinect Hardware Abstraction Layer (HAL) at the driver level. This was done so that when the RGB and real-time point cloud data were not required, both network and processor resources were freed. The other five networked Kinects sole purpose remained to create the human tracking data.

8.2 Tracking

The project used the University of Ulm's room size tracking software (RIETZLER et al., 2016) to track multiple individuals within the manufacturing cell. The software was dependent upon the use of multiple Kinect sensors each connected to a client computer, and networked to a server. The computational algorithms work in the same fashion as those discussed in section 7.2.1.1 and come with direct use of the Microsoft Kinect drivers and Software Development Kit (SDK), and as such are backed by more reliable and regularly updated software.

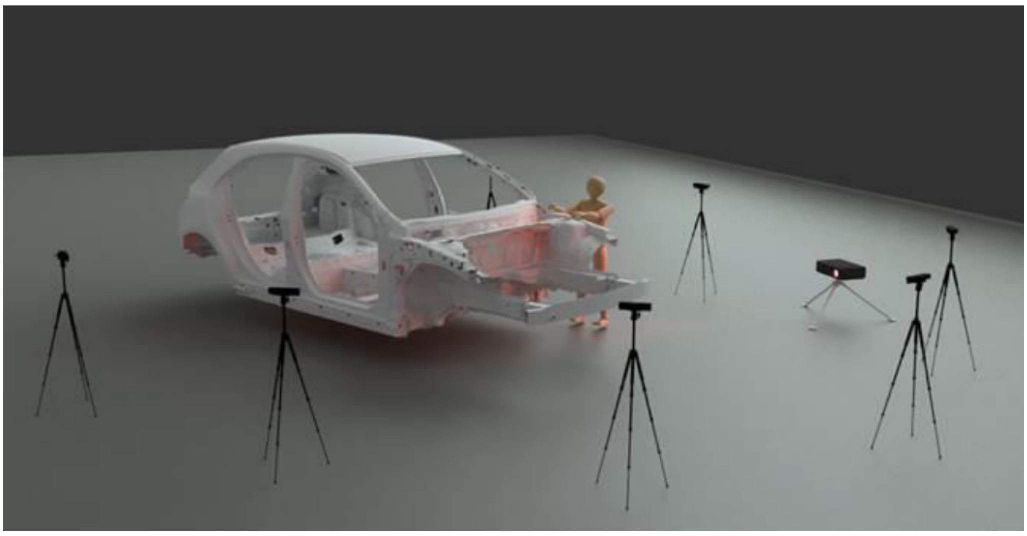


Figure 8.3: Render of the Ulm fusionkit layout (GEISELHART et al., 2016)

Room size tracking is relatively common within the design phase of high end automotive companies, and manufacturers of high value kill-tech, such a Lockheed Martin. These tracking systems are very high end, and as such area accompanied by very high costs, which limits the ability to use them in any other area part of the production cycle than the design phase. This is because the require a tightly controlled, and clean environment to function and would degrade rapidly in a manufacturing environment. This would require them to be replaced at further cost to the manufacturer. The benefit of using a smaller and cheaper system is that the set-up can be done by none experts, and the parts can be readily and easily replaced by normal shop floor workers.

Room size tracking brings many benefits over single sensor tracking as it allows the users to be tracked over a larger area. In addition, the user is tracked from multiple angles so there is a lower likelihood that the software will fail to understand which direction the user is facing (Figure 8.3). The latter difficulty arises from the fact that the skeleton data taken from the SDK is mapped to the avatar. In a single sensor or multiple sensor set up where the sensors all face in one direction, it is difficult to ascertain if the user is facing towards or away from the sensor/s. With more sensors it is possible to capture the users at more angles but even with this, it can be hard to interpret the skeleton data correctly. The solution that Ulm University arrived at was to spend more time calculating the angle between the Spine_Shoulder joint and the Neck and Head joints (3, 2 and 1 respectively in Figure 8.4). This proved effective because of the way people crane their necks forward in front of their shoulders.

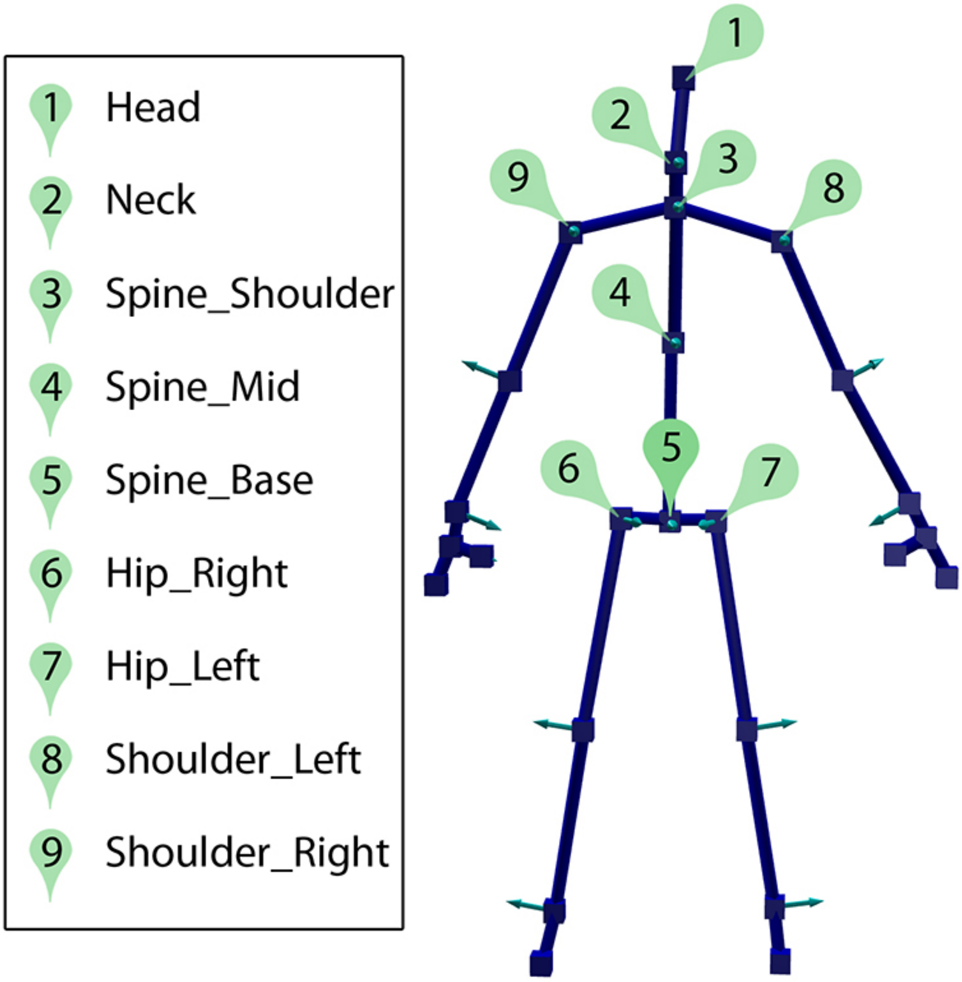


Figure 8.4: Skeleton layout created using the Ulm fusionkit (RIETZLER et al., 2016)

In addition, the same set of angles can also be used to great advantage when combining the skeleton data from the client machines with each other. Combining skeleton data is difficult for a number of reasons, the least of which is the great variation of relative locations that the majority of our joints exhibit. Our hands, feet, knees and elbows are often in constant motion and, even when relatively still, can be in a huge range of different locations relative to our centre of mass. This makes them unsuitable for use as reference joints for matching one skeleton to another. The hips, spine and shoulders all have a much lower level of relative variance and are therefore more suitable for comparison with each other. The smaller relative position and variation in orientation means that best-fit algorithms, which can be used to match each client skeleton to each other, are solved faster. However, all these joints lie in a single plane, and this raises again the previous issue mentioned concerning the direction the skeleton is facing. Instead of using the limb joints, with their high variance, it is faster to solve this problem using positional information from the head, neck, spine and shoulder joints.

This solution is pretty standard for body tracking software, up to the point where the tracking is done in more than a single plane. Most body tracking is done in a single plane and as such it is difficult to ascertain whether the tracked user is facing towards or away from the tracking hardware. The use of the head, neck, spine and shoulder joints to rapidly solve this issue allows for this to short falling to be overcome.

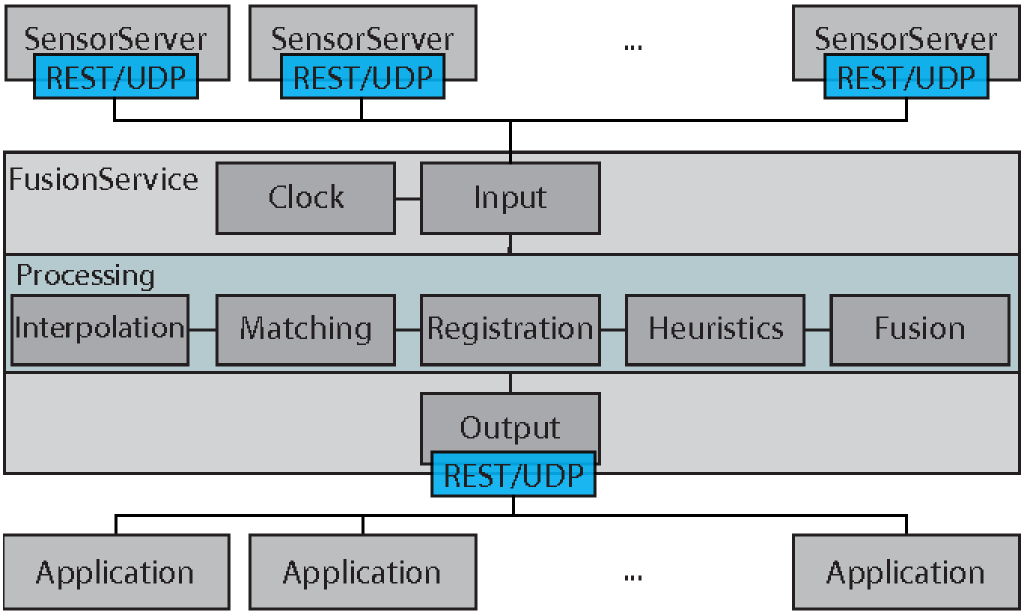


Figure 8.5: Software abstraction layers for the Ulm fusionkit (RIETZLER et al., 2016)

The combination of the skeleton data sets takes place on the aptly named FusionService server. Each SensorServer acts as a client to the server and sends time stamped data sets of the skeletal data each has captured. The data is sent over UDP as it is more forgiving with regards to packet loss than TCP/IP. This allows the FusionService to drop and ignore any timestamped frames that are missing and to continue processing the real time data stream of frames. With TCP/IP the dropped packets would be re-requested at the network level before the FusionService decided to drop and ignore them. Otherwise the re-requested packets would arrive with a time stamp that the FusionService had already calculated, forcing it to recalculate data that may have already been sent out to the application and so reducing the approximation to a real-time process. Consequently, the data sent to the Application layer would be further behind that of the user in the manufacturing cell.

Figure 8.5 shows how the server receives the data sets and fuses them before sending them back out to the Application level. It is important to point out that for use with the DAAC the protocol was changed to REST/TCP. This was so that the UnityEngine would be aware earlier if the connection to the FusionService server was lost, something which proved to be of great value while testing the system.

The tracking became noticeably less accurate when the sensors were either moved, or subject to vibration. The former could be accounted for by affixing the sensors to the frame itself. The latter was discovered when overhead cranes were used, causing sufficient vibration to reduce the accuracy of the tracking. In fact the tracking resolution of this software solution was low. While calibrating the global space of the tracking software with that of the VE, an alignment was considered good if a users hand was within $\tilde{5}$cm of the expected location. It is clear that for more nuanced and exacting manufacturing processes, the resolution is too low and should be considered disappointing considering that other infra-red technologies boast accuracies in millimeters. However given the low cost of this solution, and the potential of deployment in dirty, noisy and difficult manufacturing environments it was considered suitable for purpose.

The boundary conditions for the tracking were the extremities of the cell, which were $\tilde{6}$x4 meters. To cover this distance with just 5 kinect sensors, meant a significant reduction in the overlapping spaces, which in turn was probably a cause of the low resolution of the setup. Introducing more Kinect sensors may have increased the resolution of the system, and should be explored as further work within the project.

The DAAC set-up had an array of SensorServers, with one of them also acting as the server. This was to offload the FusionService processing load from the VE server on the Master PC. It also allowed for effective silo-ing of the tracking processes onto a separate subsystem.

8.3 Server Kinect RGB and real-time mesh

A final Kinect sensor was attached to the server; this sensor is used to provide a live RGB feed of the manufacturing cell. The depth sensor data was also used to provide a real time mesh that covered the same area as the RBG feed (Figure 8.6).
The combination of real time depth data and an RGB feed, allowed for the users within the VE to see and gain more insight into what is taking place within the manufacturing cell.

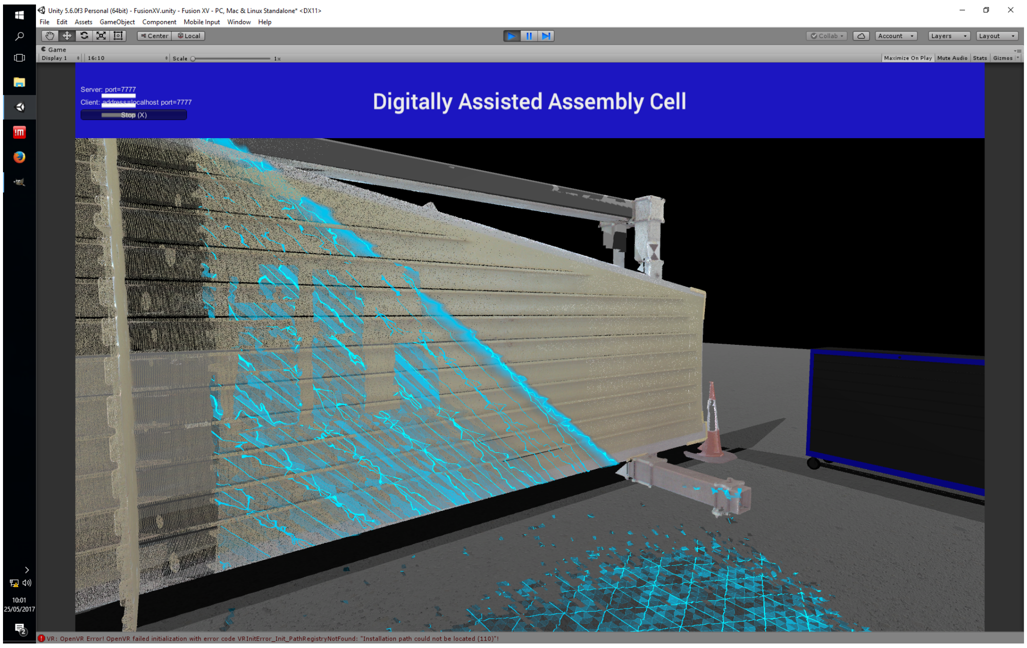


Figure 8.6: Direct import of depth sensor data from a Kinect linked to the server was used to create a real-time mesh that was visible within the VE

8.4 Virtual Environment

As previously discussed in Chapter 7, the virtual environment was created in the Unity3D engine. A simple scene was created and 3D assets were imported. The scene was set to replicate that of the real manufacturing cell itself. The majority of 3D assets were created in 3DSmax, before being exported as .FBX files. Unity3D supports direct .FBX import, but the more complex the mesh, the greater the load on the rendering engine. To help reduce the overall poly count of the scene, every asset created in 3DSmax was optimised by the following ways; reducing the vertex count as low as possible; removing all double sided faces; removing all hidden meshes and internal meshes and re-applying all material to reduce the associated meta data. For assets that were created in 3DSmax themselves, the process was less complicated. However, a number of assets came from other CAD packages such as AutoDesk Revit, Maya, and inventor (AUTODESK, 2018). 3DSmax could handle the import of CAD from all of these products thanks to its wide-ranging support of proprietary file formats. The difference between mesh suitable for games and professional CAD objects is a vast amount of data, and the stripping away of these non-pertinent data is vital if a game render engine like that within Unity3D is to cope. CAD was not the only source of 3D data, a Leica laser scanner was also used to capture the manufacturing cell itself.

The laser scan was achieved using a Leica laser scanner (LEICA-GEOSYSTEMS, 2018), mounted with a Digital Single-Lens Reflex (DSLR) camera to provide the RGB values. The scan data was taken from multiple locations across the factory floor resulting in a large data set that covered most of the shop floor area. This data was imported into Autodesks ReCap 2017 software, where it was culled into a subset that contained solely the wingskin and its mountings (Figure 8.7). The

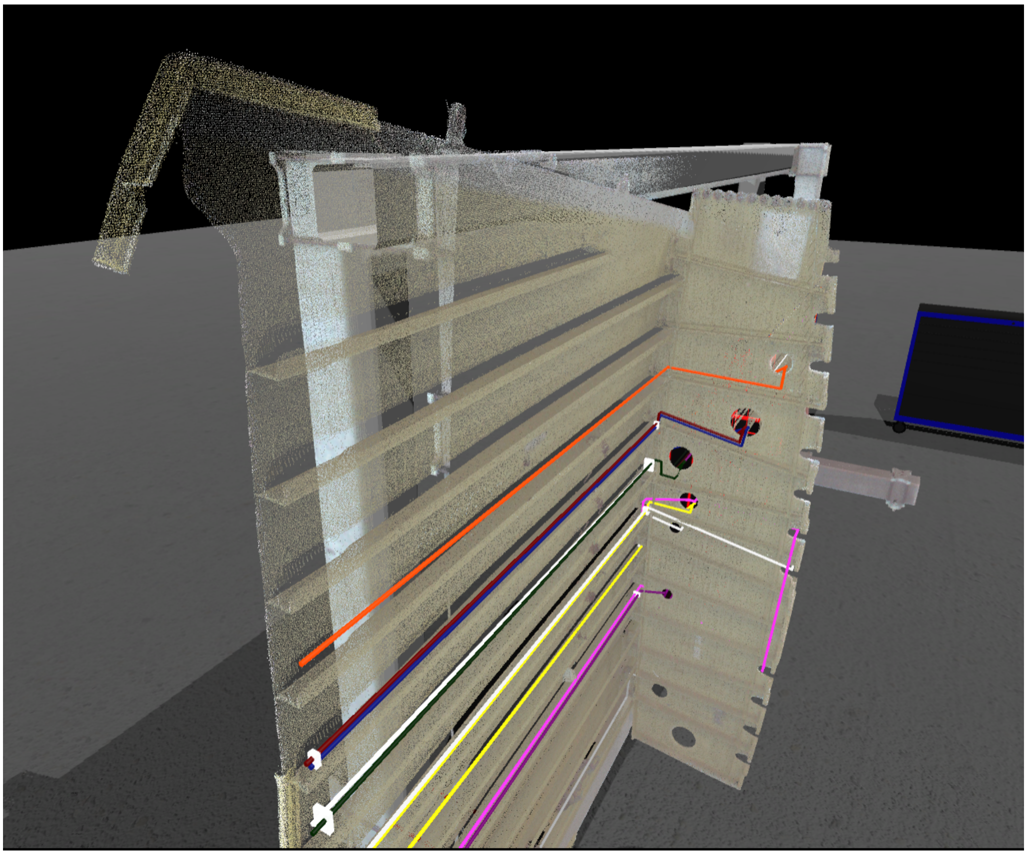


Figure 8.7: In VE mesh representation of point cloud data

resulting point cloud data could not be imported directly into Unity3D as it only supports mesh formats, not vertex formats. The conversion pipe line involved exporting the point cloud data into a Free and Open Source Software (FOSS) file format, which was then imported into Meshlab (MeshLab, 2018), where it was combined with a second separate FOSS export of the colour metadata. Meshlab was then used to convert the point cloud data from i,j,k angles and r distance from the source coordinate system into cartesian x,y,z global positions and RGB values. This was then exported into a third FOSS file format before being imported into its own Unity3D project. Within Unity3D a set of scripts created for the Framework 7 funded project Impart (Fabra, 2018) was used to convert the point cloud data into small mesh sets. A second set of scripts created a set of transparent textures that contained points showing all the RGB data. These meshes were then made children of a parent gameObject which could be exported as a separate asset. This process was completed a number of times before the wing skin could be imported into the DAAC VE scene. The result was a high accuracy and resolution digital twin of the main element of the manufacturing cell, at a poly count which could be handled even on a mid range Android device (Figure 8.8).

The wingskin was used as it was the underlying Internet Protocol address (IP) was cleared for use with the project by Boeing. Ideally a Rolls-Royce (RR) heat exchanger would have been used, but

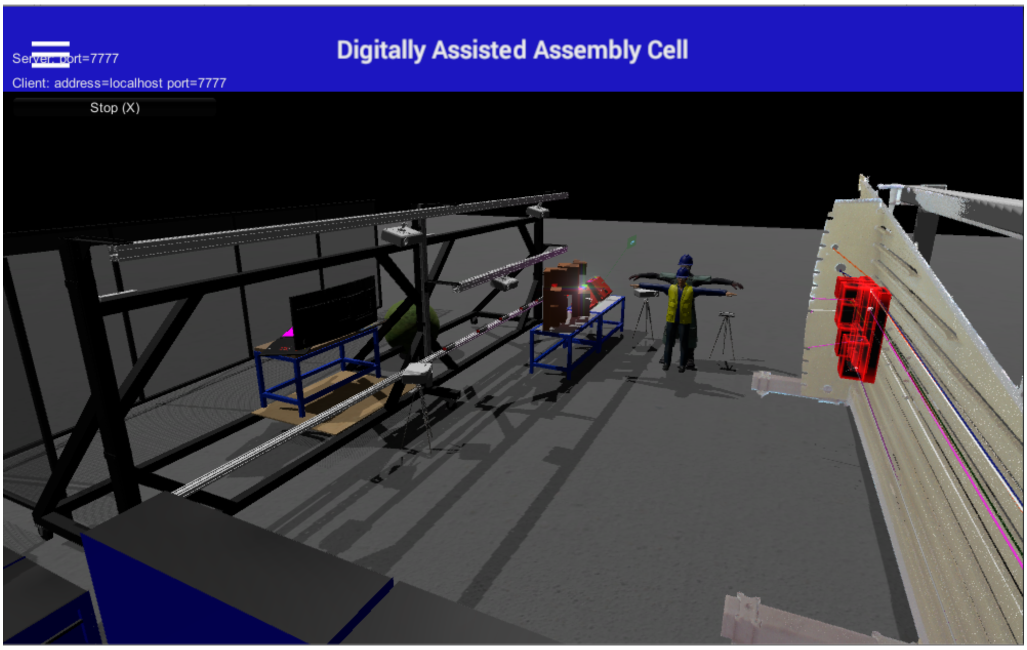


Figure 8.8: Cell Layout within the VE

clearance to use such a vessel was unavailable to the Research Engineer (RE). As such parallels must be drawn between the assembly processes mentioned, and results must be considered more generic, than specific. This may lead to some elements of simulated manufacture being seen as tenuous, and dislocated. Unfortunately this is unavoidable, and the application of the DAAC to the manufacture of nuclear components will need to be explored more fully in future work. It is requested that readers from within the nuclear industry draw what parallels they can between the manufacturing and assembly processes proposed and those more likely to be associated with nuclear manufacturing.

An essential feature of the DAAC VE was the capability to have multiple users in the same space, at the same time. To achieve this the VE would need to handle relatively complex network synchronisation processes. Fortunately a game engine such as Unity3D already has a lot of this natively baked into it. This is solely because multiplayer gaming has been such a large aspect of the computer gaming scene for so long. The DAAC project was able to harness the unity multiplayer code base, which is made available in the form of the UnityEngine.Networking class. A major challenge with having multiple users within any space is that of propagating the changes made by one user to the others. This synchronisation challenge was addressed within the DAAC using methods of event driven ownership which are further discussed later in this section.

To interact with the low level networking code that handles the handshakes and server client relationships, Unity3D provides a basic NetworkManager script. The NetworkManager script accepts a variety of parameters, the most important one being setting the game instance as either a Host server or as a Client connecting to another host. In essence, the NetworkManager is the script that

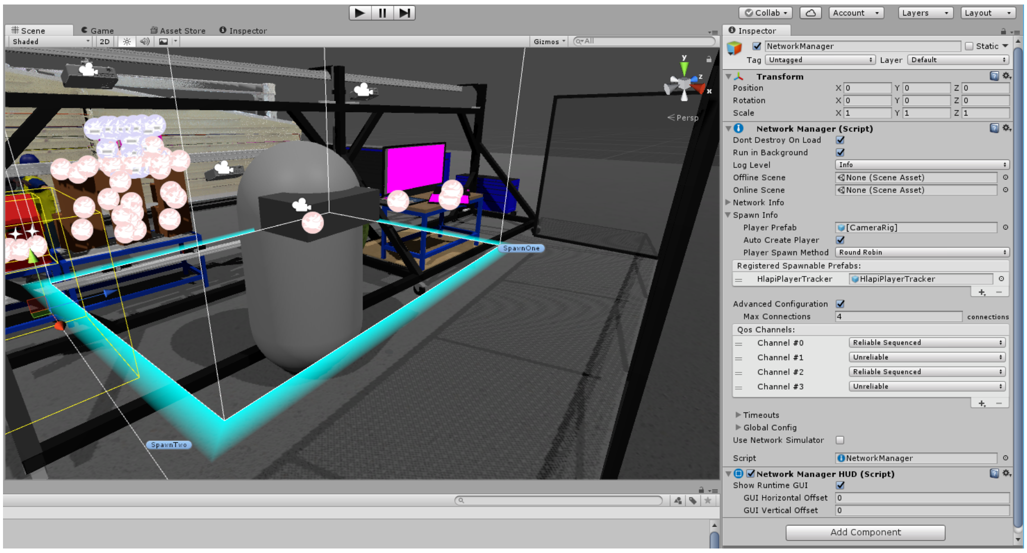


Figure 8.9: The Network Manager Script HUD, and an image of the spawn points within the VE

handles the in-network game states. An example of this is the function OnServerAddPlayer(), which will instantiate or spawn a client-owned player prefab each time a new client joins the server. For instances where multiple users are expected to spawn into the game, it is important to specify multiple spawn points. This is even more important when the users are using Virtual Reality (VR) headsets, as spawning inside of another user's headset is very disorientating.

The NetworkManager HUD is a front end in-game GUI that allows the user to choose whether to act as the host/server or join another host as client (Figure 8.9). It is a simple front end that makes calls to the on-device NetworkManager and then reloads the scene or a new scene with the new variables. This GUI was particularly useful within the DAAC, as it prevented the need for hard coding IP addresses and/or writing a custom class. In turn, this allowed for a greater amount of flexibility when testing different hosts and adapting the network.

Having the ability to designate the game instances as host and client and provide the means to connect to each other, covers some of the harder lower level network management. The higher level VE interactions and manipulations also have to be adapted. The synchronisation of events that takes place in one user's VE is essential for collaboration. The movements of the tracked users' avatars must be seen by every user within the VE, just as the changes in work instructions and laser pointers must also be seen by all VE users, as well as the manufacturing cell users. It is important to differentiate the tracked users' avatars from other elements of the VE that are synchronised across the networked VEs.

Every object that requires synchronisation across the network must have a network identity. This identifies the object as one that must be synchronised across the network. If every object in the VE were identified for synchronisation it would create a great deal of traffic between the server and its clients: in fact for each extra client the amount of traffic increases exponentially as every client must be aware of every object's location and orientation. The amount of traffic is reduced in a number of ways.

The most effective means is to reduce the number of networked objects to a minimum. It is not necessary for any static object to be synced across the network, nor any moving objects that do not affect other users. For instance, whether a tool box or drawer is open or not, probably does not matter to other users. This led to only the dynamic virtual objects being tracking with the VE. This significantly reduced the amount of data that was required to be sent across the network.

Ownership is another means of reducing the network communication. Some objects may move or change state for reasons not governed by the clients. This can be done on the server and then sent to the clients, without the need to check with them what state they have the objects in. In this respect, the server has ownership of the object, and the synchronisation is a one-way delivery of server information to the client. Conversely some objects are only controlled by a single client, such as the client's avatar. In this instance the synchronisation has the extra step of the client informing the server, which in turn informs the remaining clients of the first client's avatar location.

As only one user can interact with an object at one time, then ownership of that object can be passed from client to client as they interact with it. This prevents synchronisation issues of two clients being in conflict over the location or state of objects. At first this might seem like a limitation on the system, in real life the practical reasons for multiple users holding a single object are often due to its weight or how difficult it is to handle. In a VE neither of this issues exist, as the object has no real weight, and so any users would be able to pick and manipulate and object no matter how heavy or unwieldy it might be seem.

The tracked manufacturing cell users' avatars were not synchronised using the UnityEngine.Networking class. Instead, to reduce the level of synchronisation the game engine was handling internally, each VE instance polled the same Ulm Fusion Server for the position and orientation data frames for the avatar. This reduced the number of items the clients were waiting on the server for synchronisation and, in doing so, helped reduce the networking bottle neck of the server client synchronisation relationship.

The greater the number of objects that the game engine has to synchronise between clients, the higher the load on the server, and the more likely that clients will have to wait longer for synchronised events or other users' actions to become apparent. This can be damaging to the sense of presence experienced by the VR users. The VE scene was created with as few networked objects as possible. To further optimise the experience for VR users, the synchronisation resolution and polling rates for the networked objects was increased to the point where their movements were smooth and steady. In this way, a compromise between high fidelity networked positioning and performance was struck.

One of the most important networked components were the users themselves. The users in the manufacturing cell were represented using game engine avatars. The avatars were imported and edited in Autodesk Maya, and correct Personal Protective Equipment (PPE) and a bone structure then applied. This created an effective digital twin of the manufacturing cell user. The bone structure was created to mirror the data structure from the Kinect sensors, allowing a one-to-one mapping of the information from the sensors to the structure within Unity3D. Once imported into Unity3D, an animation rig was applied to the avatar, a process involving mapping the .fbx data bone structure to Unity3D equivalent.

The Ulm Fusion software was then configured by the users maintaining a T-Pose within the manufacturing cell, allowing the software to configure the sensors relative to each other in the real space. Calibration on a per user basis was not necessary as the distance between joints were not recorded, and the avatars were of a uniform size. Once set-up, the sensor data is combined in the Fusion Server into a global space, which has to be transferred into Unity3D. This was achieved by setting up a gameObject within the VE, which would represent the origin for all coordinates delivered from the Ulm FusionServer. By re-positioning and re-orientation of this origin point, the tracked movements of the users in the real world were mapped to the ones made by the avatar within the VE. A period of calibration is required with manufacturing cell users standing in positions that can be matched to areas within the VE.

Figure 8.10: Two models of avatars used for in VE representation of manufacturing cell users

The import of the manufacturing users through the use of avatars allowed all the users in the VE to see them (Figure 8.10). To allow the users in the VE to see each other, and the users in the manufacturing cell to be able to see the other cell users, user representations were created which displayed the other users' location and orientation. This allowed users to see where other users were and also which direction they were looking.

The user representations were created within Unity3D, and then saved as a user prefab, which allowed a new user prefab to be spawned each time a new user joined the VE. The prefab also served as the host for the multitude of scripts that allowed users to navigate and interact with the VE. As spawned prefabs, the ownership resided with the clients themselves, with only chosen elements of the prefab such as location and orientation synchronised across the network. This allowed all spawned users to make function calls from scripts that they executed as the owner.

8.4.1 User Prefab

As previously mentioned, on connection, each user of the VE spawns into the environment at a designated spawn point. The spawn points alternate between two different positions to reduce the likelihood of a user spawning inside of another. The prefab is made up of a single root gameObject, in this case called the [CameraRig], and a number of child gameObjects which make up the required parts of the prefab. The user prefab used in the DAAC project was an adaptation of one provided by the SteamVR (VALVE-CORPORATION, 2018) plugin for Unity3D. The players prefab would be the only one in the scene that had a blue material used as the mesh render, all other users would be seen by the player as per Figure 8.11. The black visor was used to demonstrate which direction users were facing. The blue square is a representation of the reach of theVR users. Within the VE it is invisible to all users, but within the editor it serves as a good guide.
The prefab hosts a number of scripts such as:

- Network Identity
- Network Transform
- PlayerController3D
- Playerset-up
- Work Instructions Networked
- GUI on off
- Hlapi Player
- Dissonance VoIP on off
- Network Grab Manager
- Pick Up Script

The code for these scripts is held within Appendix D
The network identity script, which gives the local user authority over the parent and child gameObjects is paired with a Network Transform script which synchronises the location and orientation of the user's prefab with the server, and hence other clients. The prefab also contains the PlayerControllerScript3D (Appendix D.11) similar to the camera_movement used for the S-Frame project for non VR movements.

The prefab contains a camera for each eye, allowing for stereoscopic vision, as well as a third camera that is reserved for the use of non-VR users. A Playerset –up script is used to govern which

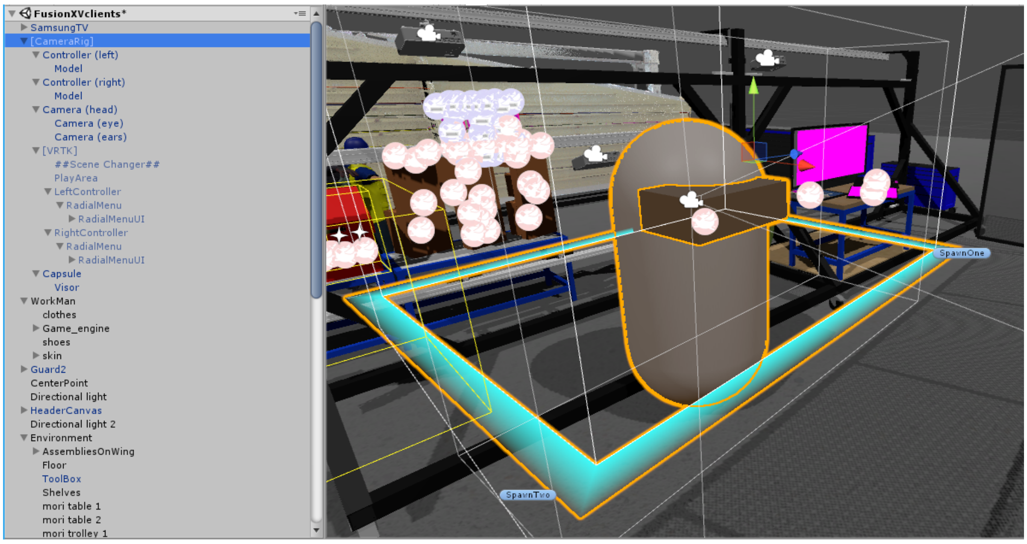


Figure 8.11: The user prefab in the sceneGraph and the VE

elements are enabled or disabled when the user spawns. This script cycles through a large number of gameObjects to ensure that other clients' prefabs child gameObjects such as their cameras, and that scripts are disabled on the clients' instance. This is required to stop cross contamination, and ensure that the correct camera is used for the user. The script uses a single array of gameObjects to disable all elements of other client instances' prefabs except for the capsule mesh and visor. It has a second array that is used if the user is not VR enabled, in which case, it disables all the VR elements of the client's prefab.

The work instructions are controlled by the Work Instructions Networked script, which uses a looping function to step through the enabling and disabling of each set of components for the relevant work instruction. These need to be synced across the network, which was done using a variety of [SyncVar]s, [Commands] and [ClientRPCAttirbutes]. The work instructions could also be fully enabled or disabled depending on what the users required. As the gameObjects used in the work instructions were already in existence on the server before the client joined, it meant that the work instructions array had to be created when the host created the game. This array was then used by all clients by referring to the gameObject it was created on, which is one way to get round being unable to create true global variables or arrays within the game engine itself (Figure 8.12).

The GUI on-off script was a simple client side script which would disable the GUI overlays. This was important for use with the projector systems, as well as for the VR and desktop users who would not require the overlay to navigate or control the VE.

The Hlapi script, part of the Dissonance VoIP system (DISSONANCE, 2018), was used within the project to allow for client control of the VoIP commands, whilst also supplying the clients location data which, if required, could be used to simulate the origin of the user voices within the VE. This particular function was disabled, as it increased the computational load on the Server and Clients,

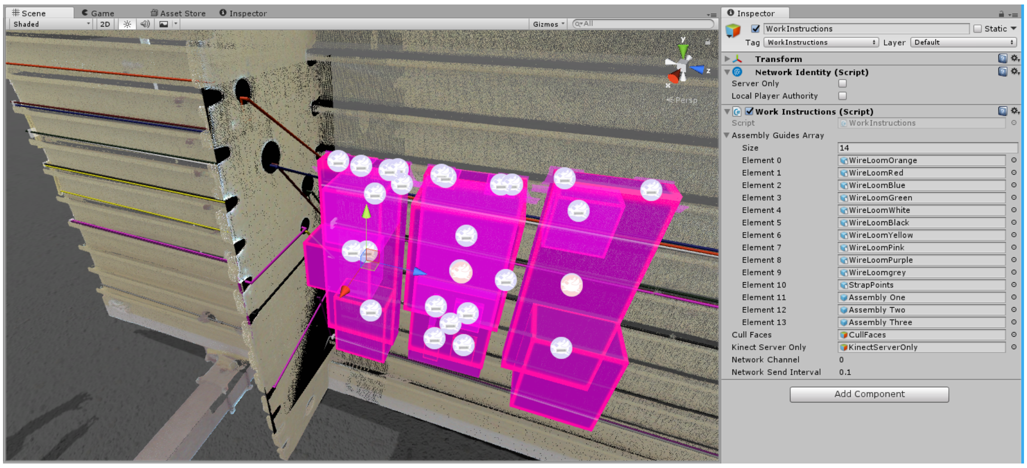


Figure 8.12: Server side script which designates the work instructions gameObject array

which reduces the synchronisation, making it difficult for users to hear each other clearly at all times.

Dissonance VoIP on-off script was created to enable and disable the voice broadcasting element of the VoIP elements. This reduces the amount of computational and network load on the system at times when the users are not broadcasting.

The Network Grab manager script was created to work alongside the Virtual Reality Toolkit (VRTK) plugin (THESTONEFOX, 2018). This script worked by passing ownership to the client of any object that the HTC VIVE handsets moved through, and which allowed the client to pick up and move objects using the handsets. Once the users' handsets were released, the ownership was returned to the host server, so that other users could also pick up and move objects. All the moveable objects contained Network Identities and Transforms.

This script was created as a work-around for the issue where users could only have ownership of objects which they spawned, the same issue that the work instructions scripting encountered and circumvented using a pre-made server array and led to the users being unable to pick-up and move objects.

To limit the number of objects that these and the pick up script would interact with, all moveable objects were assigned to their own LayerMask, thus reducing the processing load on the clients and servers.

The pick-up script worked by using ray casting from the user's mouse to identify which object the user was trying to interact with. When the user clicked on the object it would be moved closer to a screen location, and authority would be assigned to the client user rather than the host server. This allowed for the correct [ClientRPC] calls to be made. The script had another feature, which, depending on ray casting from the user, was to move the user to one of three projector positions within the VE; make their prefab material translucent; and change the colour of the projector model material to a random pastel shade. These functions were also available to be called from the GUI overlay itself or using a key press combination. The reason for these actions was to hasten the set-up process of the projection elements of the project.

8.4.2 Projectors

Each projector ran its own client version of the VE and as such the client user had to position their camera in the exact location required to project the correct alignment of object back into the DAAC. Elements of the pick-up script were designed to accomplish exactly this. Each projector had a script attached to it that held the values for the position and orientation that the client needed to be in for that projector. These values were then returned to any client that either clicked on the projector or used the GUI interface to move to a projector location. With the user's position and orientation matched to these values, the user was then made translucent so as not to interfere with the other clients, and the Field of View (FoV) of their user prefab's camera was adjusted to suit the projector. Finally the projector itself was changed to a random colour to show other users within the VE that the projector had been set up.

Figure 8.13: View of the projectors in use within the VE

The actual set-up was done over the network using RDP and NoMachine. The projectors were aligned in a cyclic process of adjustments of the projectors, the frames and the positioning of the clients within the VE. Once one projector had been aligned, the others were quicker to set-up, as the overlap between the two could be used.

Once the projector client was set-up in the right location, then a culling mask was used to remove any and all details of the VE that might obscure or confuse the work instructions. The only elements not culled from the camera view were the gameObjects that made up the work instructions, and the laser pointers. This effect was achieved by assigning the viewable objects to their own LayerMask, then setting the camera to render only objects within that layer.

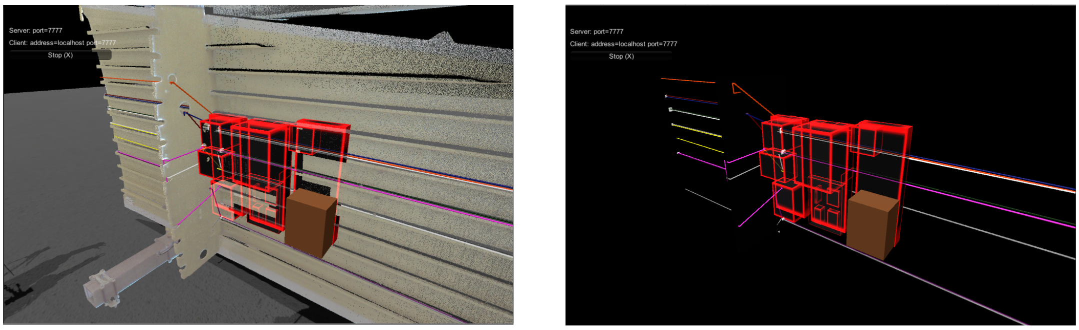


Figure 8.14: View from a projector client pre and post image cull

The work instructions projection worked by projecting a target area for the assembly boxes using a material that highlighted only the edges of the boxes. The cable locations were shown using coloured cable guides that had a pulsing lighting effect to help them stand out. The assembly box targets were used as snapping locations for the brown cardboard boxes stacked elsewhere within the VE. These boxes were given rigidBodies and affected by gravity in a similar fashion to the lego bricks discussed in the previous S-Frame project. This allowed for clients to pick up the boxes and move them into the target areas where they would snap to the correct location. In this way it was possible for the users within the VE to demonstrate working methods to other users within a real manufacturing cell.

The use of in-game physics is limited to certain components within the simulation. This is to both reduce the processing load on the hardware running the game engine, and also to benefit the users within the environment. As anyone who has lost keys behind a sofa will attest, if gravity did not act upon the keys they would be a lot easier to find, as they would simply remain exactly where you left them. The reduction on the processing load of the hardware running the game engine is so important when considering game engine physics that it deservers further explanation. Game engine physics are always very simplified versions of their real world equivalents. This is because for the most part high accuracy is not required within a game, but also importantly because in many cases high accuracy and real time visual representation is simply impossible without the use of a super computer. To save on processing power game physics engines reduce the complexity and number of calculations performed. This leads to phenomena often referred to as glitches within games. One common phenomena is that of a collision mesh being calculated less frequently than the vector calculation. This leads to an object moving within the area of another before the game engine calculated whether they have collided or not. The end artifact is that one game engine object becomes stuck within another, because it moved faster than the collision mesh was calculating for.

Within the DAAC selective application of the in-game physics engine helped to reduce issues such as the one previously mentioned. To further reduce the complexity of such calculations simple boxes were used in the assembly processes. These could be used to represent any number of actual real world components, but importantly were computationally cheaper than more vertex heavy meshes. As the DAAC is purely a proof of concept that such a system could exist, and boxes could be used to display the potential of hierarchical assembly processes.

To further aid the VE users in assisting real world manufacturing users, a set of laser pointers were created.

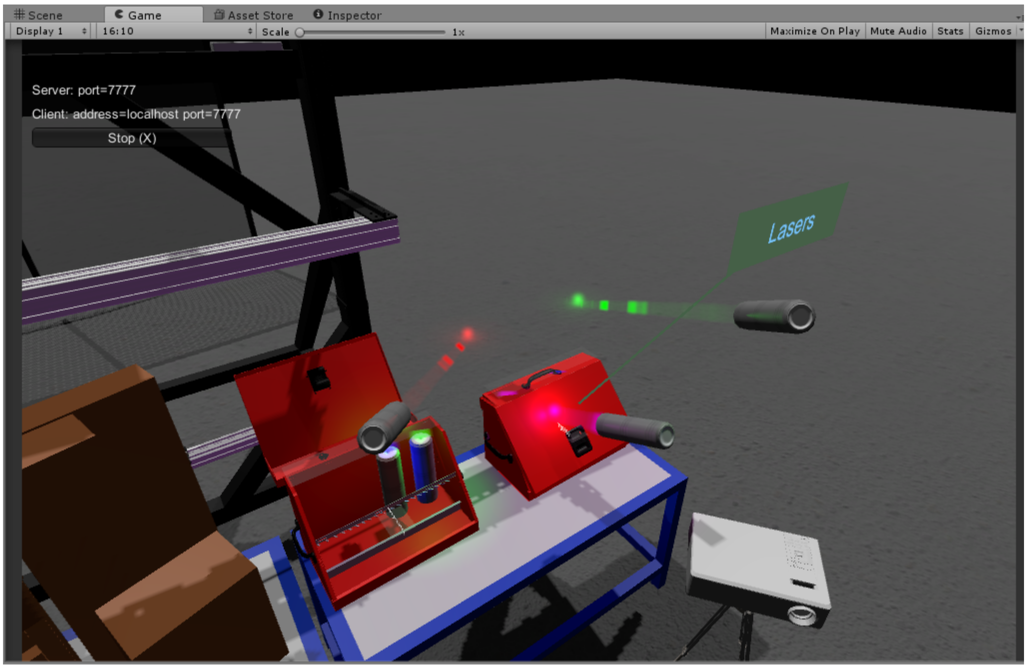


Figure 8.15: View of the laser pointers in use within the VE

Similar to the virtual cardboard boxes, the laser pointers could be picked up and positioned to assist in giving instructions to users within the real manufacturing cell. The laser pointers were not affected by gravity, so that they could be placed anywhere within the VE and left to point in the same location. Instructions such as "Move box C from the red laser location to the green laser location" could be given.

The manner in which users interacted with the VE will be further discussed in the next section, detailing the HCI methods for each platform.

8.4.3 Human Computer Interaction

Users of the DAAC have a number of methods available to them to interact with the VE itself. These are split into three main platforms: Android, Windows desktop, and Window VR using the HTC VIVE. It would be possible to use other VR headsets that are compatible with the steamVR and VRTK plugins. The manner in which users have been interacting with machines has changed drastically in the last decade, as users move from the keyboard and mouse to tablet and touch technologies, and into Augmented Reality (AR) and VR manipulation techniques. The DAAC encompasses these advances, and seeks to be accessible to a range of ability and skill. It is hoped that this allows users from a greater variety of technological backgrounds the opportunity to interact and gain value from using the cell.

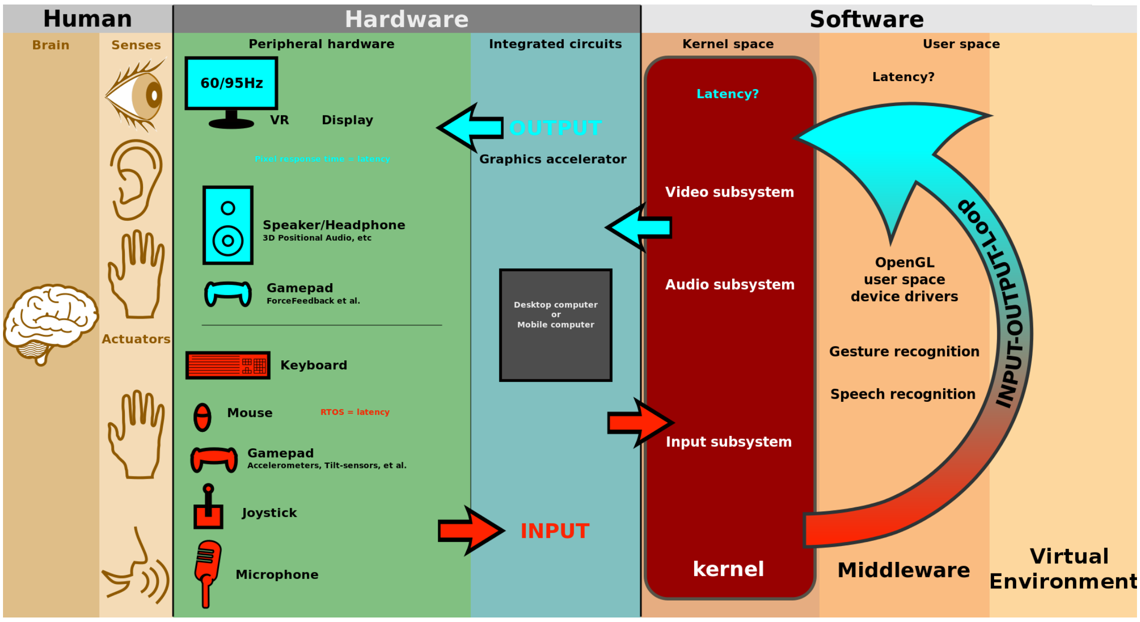


Figure 8.16: Diagram of the Human Machine Interface (HMI) adapted from (SCOTXW, 2014)

8.4.3.1 Android

For users within the manufacturing cell itself, wearing a full VR enabled headset, or having to move in and out of the cell to access a desktop machine is impractical. Instead, a tablet or mobile phone is more suitable for allowing viewing and interaction with the VE. A single ear piece allows for VoIP without reducing the users' ability to listen to other ambient sound within the manufacturing cell itself, something which could pose a health and safety risk.

In devising an intuitive touch based interaction within the VE, a menu system was created using the Material theme guidelines from Google (GOOGLE, 2018b). The principles set out a clear and concise set of guidelines for creating touch based interaction on the Android platform. The aim was to create a GUI system that any user of Android would be able to use at a glance. This involved the creation of two overlay canvases and an event system. In order to register the users' input a Unity native event system was added to the scene which, when enabled, would listen for touch based events and relay the location to the unity engine. The system also had native look actions which allowed users to look around the scene by using two fingers placed on the screen at once. In addition, users could use the finger press action to simulate a mouse click, permitting users to pick and manipulate objects within the VE.

The Header Canvas contained a simple project title, and was displayed at all times as it contained

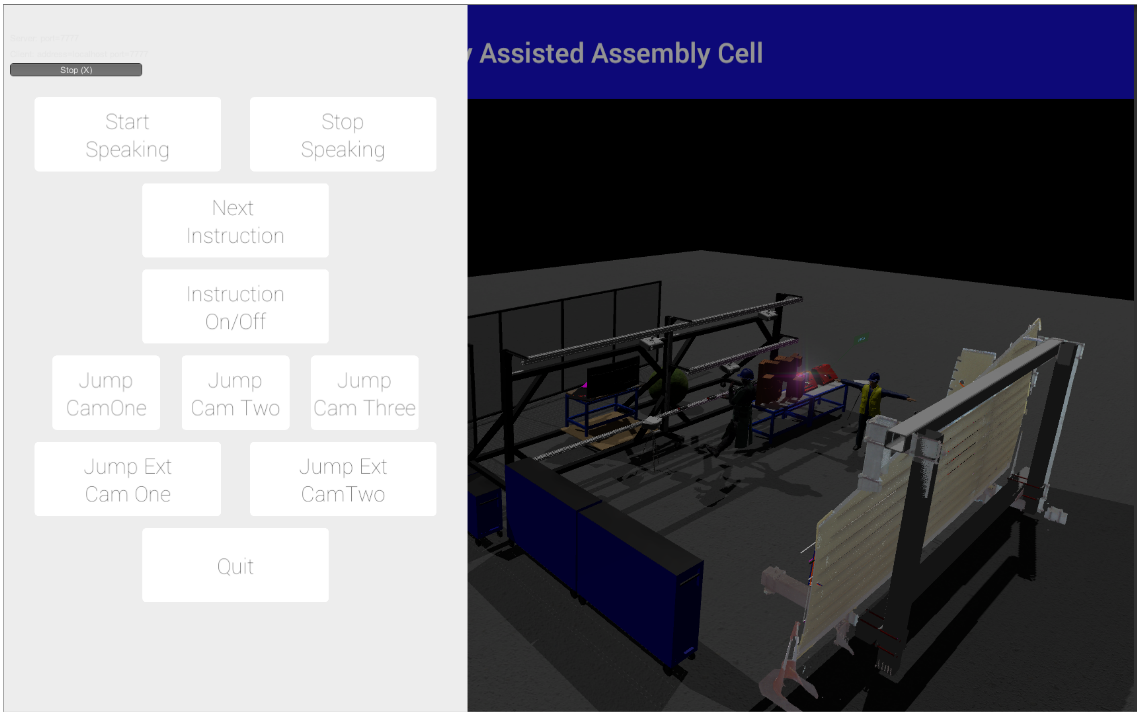


Figure 8.17: Material themed slide out menu GUI for Android

the "hamburger" menu button. The side menu canvas contained the animations, and buttons that called up their corresponding functions which were set to slide in and out from the left hand side, when the user pressed the menu button. This allowed for maximum visibility of the VE when the menu was not required. The buttons were allocated in such a way that allowed for the manufacturing cell users to have access to every feature of the DAAC.

The top two buttons enabled and disabled the HLapi VoIP gameObjects as well as the broadcasting elements of the Dissonance service. These allow manufacturing cell users the ability to toggle on and off the broadcasting of VoIP communication to other cell users, and was deemed a preferential method over push-to-talk or voice activation. Push-to-talk would have forced the manufacturing cell users to keep a button depressed, reducing the freedom of their hands to move. Voice activation was trialled but the background noise of the factory floor proved impossible to filter out at the correct level.

The **Next Instruction** button is used to cycle through the pre-set work instructions. A full set of paper-based work instructions which mirror those provided within the VE can be found in Appendix C.1. The idea of giving the users within the manufacturing cell control over the work instructions is two fold. Firstly, it would allow them to work unassisted, so even if no-one was available to provide training or other help, they would have access to the digital work instructions and guidance from the projections onto the wing surface. This could also be used alongside the recording of the users' workflows and paths within the VE for later playback and viewing within either the VE or elsewhere. The second reason for providing this level of control was to allow the manufacturing cell users the ability to skip through to parts of the instructions they were unsure or confused about.

The ability to toggle the work instructions on and off was accomplished by disabling the parent gameObject of Work Instructions. The Work Instruction's array objects were left in the same state, so that when re-enabled the work instructions were in the same part of the sequence and allowed users within the cell to resume where they had left off. This feature was added so that the cell users could quickly toggle the work instructions in situations where it was interfering with their current task. Examples of this are when the work instructions did not align with what they were currently working on or when the work they were doing meant facing towards the projectors, in which case the bright light was distracting, and in certain situations, painful to look at.

Switching between camera positions was done by teleporting the user's prefab to a set of locations. The locations were set to ensure a good coverage of most angles facing the digital twin of the cell. Positions one to three were the same as the projectors and in fact adjusted the viewpoints FoV to the same as that of the projectors. In this way a second mechanism for setting up the projectors was created. The 'Ext' camera positions were external view points looking up and down the length of the wing, which maximised the coverage of the working area.

The 'Jump' to buttons were created because users could not navigate the VE using gestures. This was due to the limitations of having a touch only based control system. Moving through a VE was found to be prohibitively difficult and imprecise. These 'Jump' to buttons also proved useful when re-aligning the users' camera positions to the straight and level. All the 'Jump' locations were indicated to the users by models of projectors, in order to help the VE users in knowing what viewpoints were available to the manufacturing cell users. Furthermore, the manufacturing cell users' prefabs showed the VE users where they were currently positioned and this allowed the VE users to tailor their interactions to areas that would be visible to those within the VE.

Finally a Quit button was added, called the exit function on the quit.cs script, details of which can be found in Appendix D.8. The script allowed users to close the open connections, and exit the application.

The slide menu allowed quick access to a range of features but also allowed the user to use most of the screen area to see into the VE itself. In this way, the Android device served as a window into the VE itself, providing the manufacturing cell users with the ability to communicate visually with users within the VE and vice versa, while the VoIP provided oral communication channels.

8.4.3.2 Virtual Reality

This bi-directional link allowed users within the VE to interact with those within the manufacturing cell. While using a VR headset, the material based GUI was disabled, as having a menu system overlaid that close to your eyes is uncomfortable, and it would limit the FoV for the users. Instead, a set of controls were built around the HTC Vives handset controllers. These handsets are tracked using the same HTC VIVE Lighthouses that tracks the headset. A 3D model of the handsets can then be created to display the position and orientation of the users' handsets; a set of scripts are then attached to the controller that calls up functions when the buttons or trackpad are pressed by the user.

The VR controllers offer a more normal way of interacting with a 3D environment, by allowing the users to pick and point at objects with their hands. The buttons and trackpads allow for a greater number of separate interaction methods and this reduces the requirement for the use of gestures or other less reliable methods of calling up separate functions.

Within the DAAC, the users were able to walk around within the confines of the VIVE Lighthouse tracked area, which limited ambulation to an area of four by three metres. To allow the users to move further around the VE, a button was used to call a laser pointer from the end of the controller. This could then be pointed at the location a user wished to move to, and the track pad pressed to teleport the user there. Within all VEs it is important to ensure that all movement is initiated by the user to avoid, or at least minimise, motion sickness.

The ability to pick up objects was achieved by calling up events when the handset model moved into a model that was in the pick-up layer. A function then assigned ownership of the object to the user, which allowed them to press a button to attach the object to their handset, in effect picking up the object. Through a series of network calls, the new location and orientation of the object was broadcast to the server, and then out to the other clients. On release of the object, ownership remained with the user until the user's handset moved out of the object's mesh, at which point ownership was re-assigned to the server. To aid in the delivery of the work instructions a variety of snap-to-fit and visual cues for objects and their intended locations were provided.

To cycle through the work instructions, and to enable and disable VoIP, a menu system was created around the track. When the user pushed in any direction on the handset's track pad, a virtual menu system appeared on the VE model of the handset. The user could then use the track pad to highlight and select one of the four radial menu items. The top menu button was used to toggle the VoIP system, which enabled and disabled the built-in microphone within the HTC VIVE headset. This system allowed the VE user to perform exactly the same tasks as a desktop user while using the VIVE handsets rather than a keyboard.

8.4.3.3 Desktop

VR Headsets have not been widely accepted, and are still far from an ideal development and testing ground for VE testing. Despite the higher levels of immersion experienced with a VE headset, it is still simpler to test and run VE on a desktop with a normal monitor. This is due in part to being able to test the VE with the same tools (mouse, keyboard, monitor) that it was created with.

The DAAC allowed users to connect to the VE using only a desktop computer, and importantly, have access to the same controls as a VE or Android user. The fidelity of those controls sits somewhere between the high resolution manipulation possible using the VR handsets and the limited touch-based interactions of the Android users.

The desktop users were provided with a WASD user prefab movement script that allowed them to navigate the VE with mouse and keyboard movements. A ray cast pick-up script (Appendix D.16) was used, which allowed the users to click on any object that was in the pick up layer. They could then move the object using the mouse and place them, or point to them, anywhere within the VE. A set of shortcut keys were created to:

- toggle on and off the VoIP,
- toggle and cycle through the work instructions,
- toggle the visibility of the GUI,
- jump the user to different camera locations including the projector locations and FoV changes
- toggle on and off the RGB and real time mesh data from the servers Kinect sensor
- toggle the camera cull layers for showing only items required for the projectors

These shortcuts were original and unique to the DAAC environment and designed to help bridge the gap between the ease of use for a desktop user compared with that of one in VR. Once the user learnt these short-cuts it made navigating and controlling the VE quicker and easier. It also meant that setting up the instances of the programs on the projector clients was quicker and easier. Keyboard short-cuts meant that the projectors could be set up even at times of high network load, when receiving real time high frame rate RDP stream from a networked machine is difficult.

To allow new users to be able to engage and control the desktop environment without having to learn the keyboard short-cuts, the Android GUI was enabled by default to give full functionality. The GUI could be disabled using a key press to give the desktop users more screen real estate if required.

A full guide to navigation within the VE is provided in Appendix C.2. This provides both a set-up guide, and also a full list of client side controls.

8.4.4 Objects

A set of objects were created within the VE to facilitate communication and understanding between the different users. These objects were designed to be interacted with by users within the VE and enhance the experience of the users within the manufacturing cell.

8.4.4.1 Projectors

The projectors' digital twins were used to show the location of the projectors and identify which projectors were in use. The location of the projectors was set by the 3D model within the VE. Whether the projectors were in use or not was shown by changing the material colour in the casing slot. The colour was chosen at random, and the function call was created when a client user jumped to the projector's location. As has been described previously, this was done by either clicking on the projector using the GUI buttons or by key press. When the client jumped to the projector, they are moved to the location described by a script attached to the projector model. Their FoV is adjusted to what is specified in the same script, and their user prefab material is changed to translucent. A full set of three projectors in use is shown Figure 8.13.

The attached script was included to speed up the calibration process of aligning the projectors, as it allowed for a numerical change of coordinates without changing the position information of the projector model itself. Consequently, the exact locations of the projectors need not be where they are shown within the VE.

Once clients had jumped to the projector, then the layer masks could be changed so that their camera only displayed the objects that users needed or wanted to be projected into the manufacturing cell. This was done solely by key press as the functionality was not required on either the VR or Android platforms. The results can be seen in Figure 8.14, with the exception of the laser pointers, which are missing from the Figure but are visible in the same layer mask.

8.4.5 Cameras & Tablets

A set of cameras were set up in the same position as the projectors, allowing for live feeds to be taken from these cameras and used as a material surface. Within the VE, this was used to show the projectors on a set of monitors and tablets situated on the rear of the DAAC support frame. The tablets were placed in the pick up layer, so that users could use them to view events at multiple angles. One of the tablets was set up so that it displayed only objects that were being projected into the manufacturing cell. In this way, the users in the VE could check what was being shown to the users within the manufacturing cell.

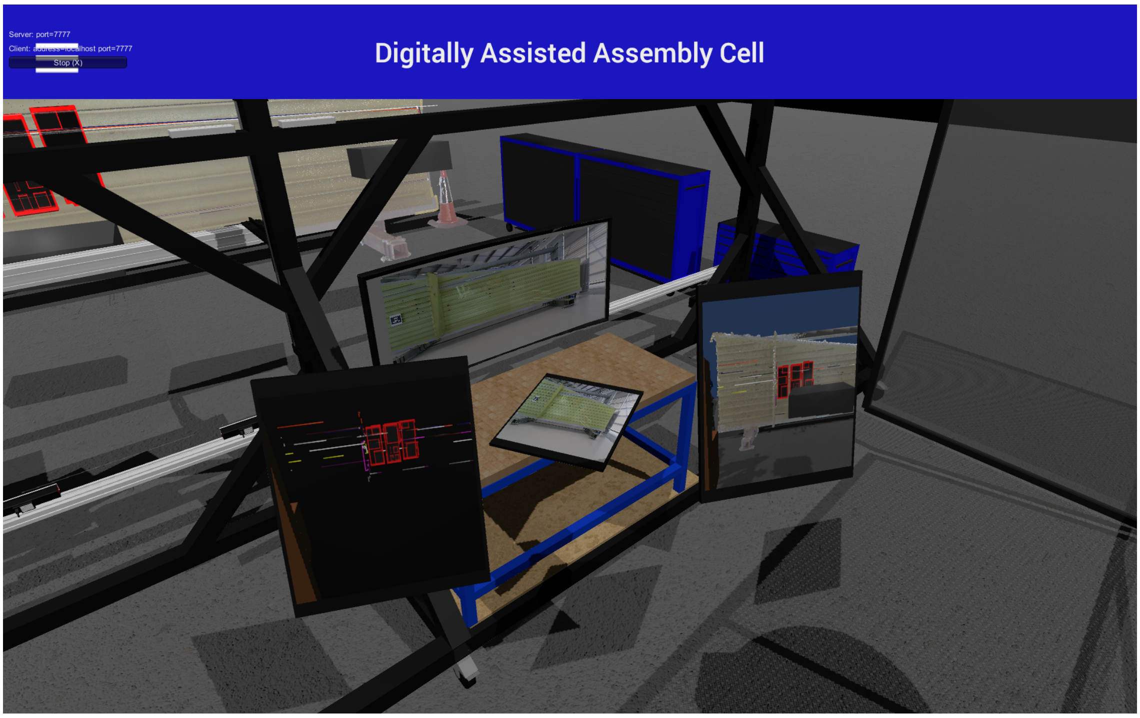


Figure 8.18: MgameObjects used as portable tablets, displaying camera feeds and the Kinect RGB feed

A final tablet and TV screen were set-up to display the RGB feed from a Kinect sensor attached to the server itself. This feed allowed the users in the VE to see a live feed of what was taking place within the real manufacturing cell. At the same time, the server Kinect sensors provided a real time 3D mesh.

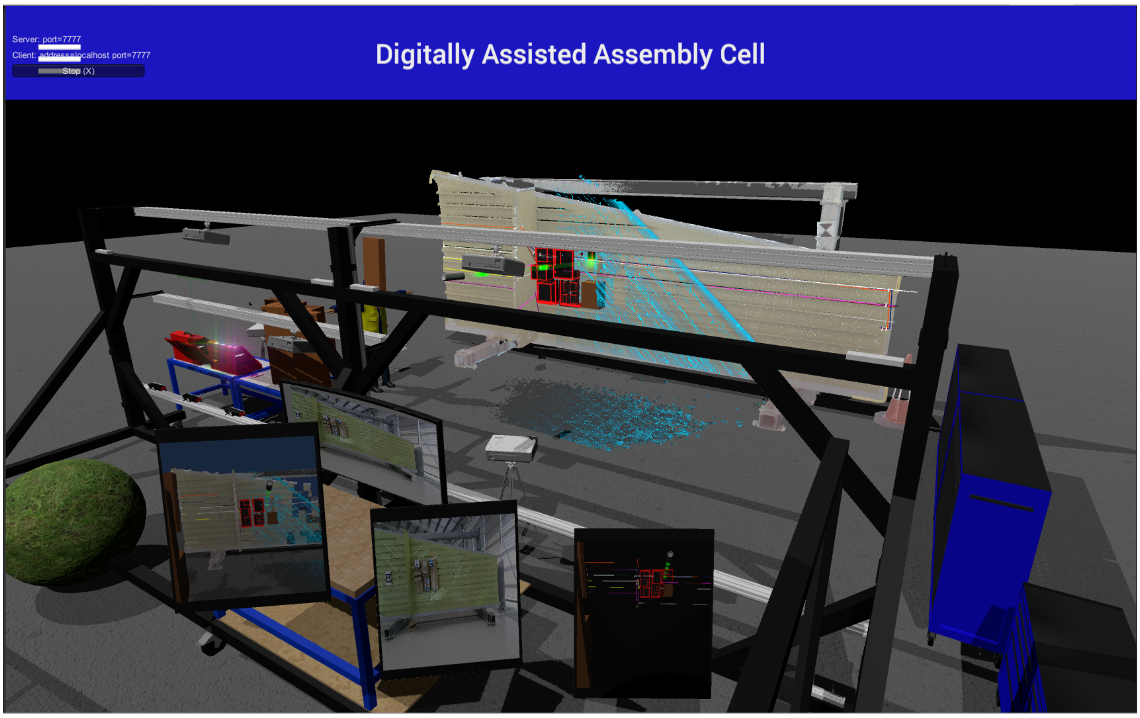


Figure 8.19: MgameObjects used as portable tablets, displaying camera feeds and the Kinect RGB feed

Due to high processor and network load caused by transmission of the RGB feed, and 3D mesh data from the servers Kinect sensor, these features were disabled by default, and could be toggled on or off by key-press.

8.4.6 Work Instructions

The work instructions could be cycled through by any DAAC user using a variety of different methods to call the functions. A few objects had additional scripting, or adaptations to assist users in communication. All gameObjects that are part of the work instructions, are in the right layer mask to allow for them to be projected out to the manufacturing cell users.

A full paper based version of the work instruction can be found in Appenddix C.1

8.4.6.1 Laser Pointers

As mentioned previously, a set of multicoloured laser pointers were created which could be picked up and moved to any location in space, with the benefit that they would stay in the exact location they were left in, as gravity was disabled on them. The lasers worked using a simple C++ script which used the RayCast functionality of the Unity Engine to create a straight line of light from the end of the gameObject to a 2D sprite that represented the end point. A new layer of objects were created, that the raycast would interact with. This reduced the number of gameObjects that were obscured when the users were pointing the laser. Two very important gameObjects were the transparent cuboids created to allow the laser points to interact with the wing skin mesh in a coherent manner. The wing skin was made up of a multitude of smaller meshes with a non-uniform surface, which led to laser interaction and the angle of the sprite being inconsistent, and at times not to appear at all.

8.4.6.2 Boxes

The boxes which were used to represent sub-assembly processes were stacked on a table next to the DAAC near the laser pointers. The boxes had their own twins set up in the locations that they were supposed to be attached to on the wingskin. They were given a flat brown material texture, and their target counterparts given a red edge mesh material. Each box had a number of scripts attached to it, including the same `snap2.js`, which changed the material of the boxes to green when they were aligned with their correct counterpart target. On release, the part then drifted towards the target and reorientated itself. Spring mechanics and dynamics then held the box in place. This was to increase the ease with which users on the desktop or Android platforms could demonstrate the assembly process.

VR users were able to move around the cell with more natural movements, and for them, the `snap2.js` script unnecessary. Instead a set of scripts were used to create target locations for every single box. These target locations would appear with a highlight of the boxes outlines when any box was moved close to them. Any box could be attached to any target location, picked off again, and re-positioned in another location. This allowed for the VR enabled environments to act as a simulacrum of the real, and they could be used to teach users the work instructions themselves. Practice sessions could either be done by users on their own, or alongside a pre-recorded avatar in the manufacturing cell. Use cases for the DAAC will be covered in the next chapter.

CHAPTER 9

Discussion

The previous two chapters provided information on how the Digitally Assisted Assembly Cell (DAAC) functions. This Chapter will cover a few of its possible use cases, its performance in testing, improvements that could be made in future iterations of the project. It finishes with a discussion surrounding the challenges facing nuclear manufacturing in the UK.

9.1 Use Cases

The DAAC was designed to be integrated into the manufacturing processes of a factory. It aims to bridge the gap between Cyber Physical System (CPS) and human production. Its use can be extended beyond that of just a manufacturing cell. It could assist with:

- training manufacturing cell users
- capturing best practice among existing experts
- integration with CPS
- data capture for use within the 'quality chain'
- cross-skill the work force
- rapid reconfiguration of the work environment and/or production routing
- remote access to the manufacturing environment

It should be mentioned that all of the above use cases are theoretical and, as a proof of concept, the DAAC should not be considered production ready. A full proto-type would require rebuilding from the base upwards, a number of recommendations on improvements are contained in the following chapter. The current proof of concept has been used to show that such a system is possible, and that the different systems can be linked together to produce a digital twin of a manufacturing cell which could be used for any or all of the points listed above. The statistical confidence that this could take place, simply does not exist as the system would require extensive testing both in beta and development phases. This is even before such a system could be tested within a real manufacturing environment, a standard progression up the ladder Technology Readiness Level (TRL) should allow for a version of the system that could be used to test the assumptions and ideas discussed in the following chapter.

As such there is no statistical confidence that any such system as this could deliver on its potential, it is the firm belief of the Research Engineer (RE) that this system or a similar rebuild of it could offer the following improvements.

9.1.1 Training

Perhaps the most obvious additional use for the DAAC is training workers in production techniques and practices. This can be done either using the bi-directional links to train users in the real manufacturing cell from the Virtual Environment (VE), having all users train within the VE, or by demonstrating in the real manufacturing cell for users within the VE.

9.1.1.1 Training in the real manufacturing cell

By allowing experts to connect from remote locations, the need to move resources and personnel to a training location is removed. Skilled experts can login from any remote location and assist in training staff in-situ. This has an obvious cost based benefit related to travel and saves the cost of renting or running a training centre itself.

Perhaps less obvious, is the benefit that personnel are being trained in the same location that they are going to perform the work itself. This allows for all local variation and nuances to be understood and accounted for by both the trainer and the trainees. Space restrictions, tool variations, service availability and environmental variables can all be taken into account by the trainer, who can tailor the session to both the trainees, their skill level and their specific location.

The ability to train remotely will also allow increased utilisation of training staff. Without travel times, the trainer can be remotely involved in multiple sessions in a day if needed. Furthermore the trainer can re-connect with any DAAC whenever further assistance or clarification is required. This could further reduce halts or delays in production characterised by the Andon system of Toyota's Jidoka methods (Toyota, 2018). With the ability to refer to an expert, the manufacturing staff could gain assistance immediately, which could help reduce the need for on-site expert supervisors.

Remote assistance from the virtual to the real manufacturing cell makes the idea of a centralised manufacturing knowledge base more practical. It can link the shop floor workers to the design and manufacturing experts who created the processes by not only allowing the workforce to rapidly clarify and solve their difficulties, but also by bringing the designers and expert engineers closer to the "coal face". The benefit here is that issues that have not been obvious until the manufacturing stage can be shown and demonstrated with greater ease to the designers and engineers who created them. An example of this could be simulating a new assembly process such as new cable routings within the VE before it is attempted in the real world. The RE created a number of simulations for complex lifts within factories for Rolls-Royce (RR) in previous Virtual Reality (VR) simulations that helped to address issues surrounding, crane use and positioning, spotter locations, and floor space required. Catching issues earlier can help by removing layers of personnel involved in the issue entailed by using an escalation process, while also demonstrating the issue with greater clarity to the progenitors of the process.

9.1.1.2 Training in the virtual

The need for the process creators to understand what the manufacturing techniques require in terms of both equipment and personnel can prove instrumental in creating better future designs. Toyota used to have engineers work on a factory floor for a full six months before allowing them to start work within a design department. As the philosophy of design for manufacture is gaining momentum within industry, so is the requirement for designers to be able to more rapidly realise their designs and manufacturing techniques. Traditionally this was done using models and mock ups and more recently through the use of Computer Aided Design (CAD) and simulation software. Examples of VR and Augmented Reality (AR) being used are less common and are predominantly in companies that specialise in low volume, high value manufacturing such as members of the military industry complex.

The DAAC allows designers and engineers to create and test work instructions; this can be done both independently or in collaboration with staff on the shop floor. Users can either be fully immersed within the VE using VR headsets or can view the process from desktop or Android enabled devices.

Once the design phases are completed, then the VE can still be used to train the work force in the manufacturing techniques and processes themselves. Using the room sized one-to-one tracking of VR, the workforce can be trained in a simulacrum of the real manufacturing cell itself.

Just as for training programmes in the real world, training in the virtual can reduce costs and allow for a central location of expertise, but with the addition that staff trained within the VE would consume less resources and require less supervision.

Training the workforce in a virtual world creates a direct feedback loop between the workforce and the design engineers, allowing for further iterative improvement in the process. If this is performed before capital expenditure on the manufacturing techniques, it would lead to even further cost reductions. Catching manufacturing issues before production is started would lead to significant savings on the cost of adapting existing processes.

9.1.1.3 Optimisation

Using the DAAC for training in either the virtual or the real world, or creating training programmes involving both virtual and real components, could lead to significant cost reductions involved in testing the cell layout and improving the manufacturing process. In addition, cost savings could be made to the more variable human components involved in the training process, such as assisting and monitoring the employees understanding and ability to fulfil work instructions accurately and in a timely manner. This would allow for more rapid testing and design iteration at both the design and deployment stages of manufacturing. The remote aspects would allow for multiple design teams in disparate locations to collaborate with each other and with the shop floor workers. This would reduce the traditional divide between manufacturing engineers in the office and those on the

shop floor.
Staff working within the VE could be trained and re-trained depending on current best practice. In addition, new best practice procedures could be created and trialled within the virtual before being introduced to staff working in the real world, improving on previous best practice for staff in the VE.

9.1.2 Capture of Best Practise

The DAAC could provide a number of methods by which best practice can be captured:

- User tracking recording
- RGB camera recording
- direct feedback by Voice over Internet Protocol (VoIP)
- simulation through VR

Movements of users could be recorded within the DAAC and viewed and replayed to other users anywhere and at any time; scenarios displaying the fastest completion of work instructions; the fewest issues/problems encountered; patterns of most efficient human movement and patterns of the least repetitive actions could all be kept for posterity

Users can be tracked in the real manufacturing cell, where a team in one factory may be completing a set of work instructions faster than others in a different factory location. The movements and actions of the faster team could then be re-projected to the slower team. Equally, the faster team's movements could be analysed within the VE where future improvements could be tried and tested.

If processes are recorded within a real working factory where the few issues/problems have been encountered, then best practice could be captured from live production sites and used to improve the quality of outgoing products.

Repetitive tasks could be minimised in both the VR and real parts of the DAAC by studying the human movement of workers, which might lead to a reduction in injuries through repetitive strain or bad lifting practices. It could also lead to an increase in concentration on tasks, due to a reduction of their repetitive nature.

Through recording of best practice, an improvement made in any DAAC could be transmitted to any other location where similar procedures exist. Best practice procedures could be returned to the design engineers to be incorporated into their future work. Best practice could also be created and reviewed solely within the VE, which would allow for the creation and dissemination of best practise before any of the work in the real world was accomplished. The creation and saving of best practices within the DAAC could leverage the advantage of scale by capturing the best manufacturing practice from multiple sites. The improvements in any one procedure could be trialled and adopted by all the others, without the need for anyone to travel between sites.

In addition, implementation of best practice could be carried out by the creators teaching others through the use of the VE itself. In this way less information would be lost in the transfer of knowledge from user set to user set.

9.1.3 Integration with CPSs

The 4^{th} Industrial Revolution (IR4.0) was built upon the creation and ability to disseminate across different systems to generate useful information. Within manufacturing the goal is to use this information to inform decision making processes by staff or machines such as CPSs. In this way manufacturing processes can be further optimised across machines, factories and as far as the value chain stretches.

The DAAC supplies what has previously been a missing piece to this network; the human element. The human element, whether that be Human Computer Interaction (HCI), assembly times, or ergonomics, has always been limited to the point of contact between human and machine. Existing solutions offer remote assistance through video displays or even laser or projector overlays. However, the combination of tracking a person's movement with assisted instructions adds an important new feature; how someone works within a manufacturing cell can be analysed with similar tools to those that are used on robots. Optimal performance pathways and time per task can be looked at with a more complete level of context.

With data sets surrounding worker utilisation, a greater fidelity of actions such as the KanBan system can be accomplished and allows for increased interaction with Material Resource Planning (MRP) systems. Using this system the delivery of parts and goods to a cell worker could be done in line with the route the worker would be taking through the cell at the time. Furthermore, goods can be delivered and removed at an appropriate time of use, reducing the level of inventory on the factory floor and increasing the working space, which will make accounting easier and increase the safety of workers. In this way, both the time and use of materials can be further optimised. If multiple DAAC systems are installed in a factory, there are increasing benefits of scale, with aspects such as factory routing being optimised and known future human pathways incorporated to create safer and more efficient factory floors.

The integration with CPS also allows for the creation of data sets indicating where in the work instructions workers are, which can then be compiled into dashboard systems used by management. In this way the Enterprise Resource Planning (ERP) systems used by management reach a similar level of fidelity concerning workers progress as they do on machine times and spindle speeds. It can be made obvious to Personnel Management how many hours, what shifts, and where individuals are working within a factory, which could lead to better optimisation of shift patterns and work loads.

The analysis of ingress and exit points and times within a manufacturing cell allows for higher level of fidelity in factory flow optimisation. The increase in flow modelling of a single factory will affect, and can be scaled to, the production chain of several factories. This can produce gains across the entire supply chain.

This will allow managers to have access to information from the ERP level all the way down to individuals within a single manufacturing cell. This could create a direct link for data, analytics and communication from the shop-floor of a factory to any level of management within a company. This could lead to a faster and more effective decision making environment. The optimisation of worker utilisation could lead to significant challenges for the workers, and could well be a hard sell to a unionised workforce.

9.1.3.1 Privacy

The author would like to take the opportunity to acknowledge the ethical and moral issues that are raised by this level of tracking. Such a technology must be tightly regulated and controlled lest we degrade the work experience itself and cause worker's embarrassment or make them feel continually under surveillance. Utilisation and optimisation of human work patterns should be centred on the well-being of the worker, as well as consideration of increasing production levels.

9.1.4 Quality Chain

Quality has increasingly become a key element in modern manufacturing since the introduction of the Toyota way, lean manufacturing and Japanese dominance in the 1970s. The heightened awareness and use of quality has led to the use of the quality chain. This is a series of documents used as proof of process and work completion. The DAAC is able to provide a window into the manufacturing cell, which allows for the creation of new items that can be used within a quality chain. Each work instruction now has, not only paperwork describing how the task is supposed to be accomplished, but also the live RGB feed and human tracking of how and when it was achieved.

This means that issues can be tracked back to the exact process where something may have gone wrong and detect whether a mistake was made, or if the current process is unsatisfactory. This should take a lot of the guesswork out of tracking, down to the root cause of faults and problem issues. With the ability to record and take snapshots of the process taking place, it will be easier to identify which process led to the creation of a fault that is manifested later. Ultimately, this will allow for greater confidence in a company's own quality chain and greater confidence in their supply chains, if every supply chain were to open up their processes to inspection by potential customers. This ability might allow for the greater sharing of best practice across companies. However it should be noted that the benefits of this root-cause analysis would only apply to non-codified processes. That is improvements discovered through this methodology could only be applied to processes not set out in codes such as ASME VIII (ASME VIII is the code associated with the rules for the construction of pressure vessels, whereas ASME VII is the recommended guidelines for the care of power boilers).

Confidence in a company's quality chain can help build better safe-guarding procedures, which can lead to significant cost savings, as well as increasing the safety of staff. By allowing for the tracking of unsafe practices through the use of a bi-directional link, the shop floor staff can record safer working methods as they develop them and communicate these to management and other workers; sharing these across the company is relatively easy, since they can be viewed within the VE.

9.1.5 Cross Skill

The DAAC has the ability to deliver work instructions directly into the manufacturing cell, both in-situ and on call. This could allow for unskilled workers to perform tasks that they had previously lacked the training to complete, with the added benefit of an improved quality chain to ensure that standards had been met.

Cross-skill in a workplace allows workers to perform a greater variety of tasks which can reduce boredom and burnout. It could improve shift scheduling, and reduce periods of down-time that might result from workers who had limited ability and lacked flexibility across several tasks. An example of this would be a machine operator who can only work when their machine is required in the process, which could only be for half of the hours of their shift. With the delivery of exacting work instructions straight into a cell, the machine worker can now perform assembly or learn other tasks with minimal re-training time. This increases the workers utilisation, which reduces overheads, and can also reduce the overall production time.

The cell also allows for skilled workers to train and assist less skilled workers, either solely within the VE or through remote assistance. In this manner the most efficient and skilled workers can assist in the training of any workers anywhere within a company.

9.1.6 Rapid reconfiguration

The projection and control of the work instructions allows users within the VE to amend the process as they see fit. By cycling through the work instructions the users can review what changes need to be accomplished in that day, and schedule the tasks as they see fit. This gives the manufacturing cell users control over the tasks, allowing for rapid changes in the manufacturing process depending on external influences, such as part delivery times, required re-work, testing new processes, or fitting in processes before shift changes. The reconfigurations can be watched and reviewed by users within the VE.

These reviewed changes can then be submitted to the DAAC as changes in the work instructions. As new methods are discovered by either workers in the real manufacturing cell, or in the VE these can be rolled out to any and all instances of the DAAC, allowing for the rapid reconfiguration of the work instructions and processes almost as soon as the improvements are discovered.

9.1.7 Remote Access

The ability to "dial in" to the DAAC from any location in the world, has a number of benefits. Already mentioned are the possibilities of:

- sharing best practice across a global network of factories
- the rapid dissemination of process change
- remote assistance when problems arise
- a centralised knowledge base of experts assisting globally
- management oversight

- the linking of engineering and design staff with factory floor staff and processes
- the inclusion of DAAC data into MRP or ERP systems

Another advantage is a permutation of remote assistance; if the DAAC was used for repair or maintenance work instead of manufacturing, then the ability to have an expert "dial in" and assist a repair technician on site could reduce the turn-around time significantly. The number of on-site specialists could be reduced, with a central bank of specialists assisting more generalist technicians remotely. Specialists would be able to, not just talk technicians through the process, but also see what they see as well. Furthermore, they would be able to tailor the instructions to the specific use case. This may require a different set of VEs depending on the issues. For businesses that rely on in-service revenue this could be particularly lucrative.

9.2 Performance and Challenges

The performance of the DAAC was very dependant on the particular case used. Despite the best efforts of the RE the computational load on the server and the VR clients was always high. This was especially true when the extra Kinect sensor was supplying RGB and depth sensor data to the server.

Although efforts were made made to cull as much data as possible and thus reduce the amount of data being shared across the network, there were occasions when it formed a bottleneck. The lower the number of clients, the better the VE performance, as less user data and fewer interactions required syncing.

9.2.1 Virtual Environment

The use of Unity3D as a virtual environment proved to be effective. It provided a high enough positional resolution to allow users to interact with each other in the VE and within the real manufacturing cell. The graphical resolution was also clear and concise, being human readable and understandable at all times. The sense of immersion when using VR headsets were high, and the one-to-one scaling allowed for normal human movements and interactions.

The object clipping was alarming for users, as the avatars and other objects moving through the cell could cause a sudden occlusion of vision. This was most alarming when the objects came from behind a user, as there was no foreshadowing and, all of a sudden, the user's view could be fully occluded. This 'trapped-in-the-object' issue has been well addressed in other applications by forcing the object to become translucent when it starts to occlude a user's Field of View (FoV). Aside from this, the experience within the VE was easy and the controls intuitive. Object movements were free and easy, the snapping worked effectively and objects were seldom lost or stuck out of reach.

Whereas the experience from the user side was relatively smooth, the majority of issues with using Unity3D arose in the creation of the VE. Unity3D is predominantly an independent game development platform, and as such, it lacks a lot of the polish associated with high end titles, and is orientated towards smaller developers. It has not been built to handle large file sizes, whether this be textures, meshes or data sets.

As a game engine, Unity3D is not designed for importing professional CAD. The support of the .FBX format allows for the creation of a work flow pipeline, requiring all CAD data to be imported via a program that can convert it to .FBX. This necessitates the loss of a great deal of metadata, including material properties. The low poly count requirement also requires a reduction in the shape and form resolution. These reduced meshes perform better in the render cycle but restrict the ability to perform object-to-object-fitting simulations. This issue is compounded by the requirement to create separate collision meshes to the objects. Since collision meshes are used in the physics calculations, the greater the complexity they have, the more computational load they create. As such, they are normally simplified versions of the visible mesh. These restrictions forced the further use of snap-to-fit points, rather than collision detection, which would have allowed accurate object placement.

The loss of meta-data associated with the models restricted the usefulness of using the CAD of components. Thus, the ability to use part numbers or material data is not present in the DAAC. However, it would be possible to export this kind of meta-data which could be re-imported later with scripted variables attached to gameObjects. This could increase the functionality and benefits of integrating the DAAC with MRP and ERP systems.
Unity3D is not designed to interact with external data sources except for data tables such as high scores. Even then, the interaction with the external data tables is limited and infrequent. This necessitates the creation of complex scripts to convert the data sets into formats that can be imported and utilised within the environment. The better scripts will do the majority of the processing as a separate thread from the main unity engine. This alleviates some of the computational load. However, at some point, the data needs to be incorporated into the render cycle, at which point the computational load can increase significantly. All of the DAACs external data sources are managed in this fashion, with as little data being handled in the main thread as possible. However the RGB feed and real-time mesh from the servers Kinect sensor still create a large throughput of data sets that need to be handled by the main thread. This is why the performance suffers when this feature is enabled.

Another feature that caused significant performance issues was the use of real-time lighting. Real-time lighting can significantly increase the level of verisimilitude and therefore immersion for users. Unfortunately, Unity3D's ability to render real-time lighting was poor. This meant that all of the lighting sources had to be baked into the scenes. This is the reason for using a sprites as the point of the lasers and other optimisations. The effect his has on the user is that the environment feels more simulated and less realistic. This can reduce the Place Illusion (PI) and Plausibility Illusion (PsI) experienced by users, and in turn also reduce the effectiveness of the training. It also has a more immediate effect on younger generations who are used to high end graphical clarity on game systems at home, and at the cinema, so notice short-cuts and short falls within graphically simulated environment even more than their predecessors.

One of the many benefits of Unity3D being an independent platform is the great number of supporters building extensions and modules to the functionality of the engine. This allowed the RE to import the point cloud data, add more complex textures and access VR support, among many others. On the other hand, this benefit has the drawback that any program using modules built by different people can suffer from; fragmentation and poor optimisation. Some of this is alleviated once the VE has been compiled as a lot of the functionality is on the creator side, and other aspects are baked in and/or controlled by the runtime engine itself. However, a large variety of modules working within the editor to ensure a level of harmony and stability, posed a real challenge.

One area that was native to Unity3D is its multiplayer networking. Coming with the promise of being easy to include and implement within a project, it proved to be challenging to add even basic functionality for multiple users. There are a number of user-created modules that offer greater ease of use, and or functionality. However, the RE was restricted by the requirements of security: all data must be handled on servers or computers owned by the RE. This mean that the DAAC must be able to run on a secure internal network without a connection to the web. The final solution is very robust, and arguable even more secure due the reduction in third party coding solutions. The functionality could be extended to any network, including ones with access to the wider web. Which would allow people to "dial in" from any location in the world.

The greatest challenge with the networking interface was ensuring that all users could interact with all the objects. This meant frequent ownership changes [Command]s and [RPC] calls. Every script that had been created for a single user environment had to be re-worked. A great deal of extra functionality had to be added to the user prefabs, which led to a great deal of script bloat. The solution often meant scripts were run on server load that would populate arrays, set ownership, and hold the required "global variables". The ownership of these would be changed by the server on certain scripted Events, at which point, the client would change variables and send the changes to the server, which would then send it to other clients. This was a sub-optimal and inelegant solution but it was functional. Further work could simplify a great deal with the creation of a C++ Class, which would contain functions optimised for user interaction in a multi-user environment.

Another area that led to user prefab bloat, was the requirement for users to be able to interact with the VE through VR headsets as well as desktop and Android clients. This meant that every user prefab carried all the functionality required for a VR user. This was disabled if no VR headset was detected. This was one of many functions that were not natively provided by the steamVR plugins (VALVE-CORPORATION, 2018). Thankfully, a group of developers were working on the Virtual Reality Toolkit (VRTK). By working closely with them and assisting with the development, it became possible to add a lot of required functionality. This included, but was not limited to:

- hand-held menu systems
- event-based collision modelling
- network enabled user models
- dynamic gameObject snapping
- multiple user cameras

The challenge of working with the group came from using a rolling release. This meant that patches and updates were added to the plugin on a daily, sometimes hourly basis. Although most updates improved functionality, it was not rare that an improvement in one area would lead to problems in another. As the deadline for the project came closer, the RE had to freeze the updates on the plugin to avoid any issues further down the line.

Although the issue was most prevalent with the VRTK plugin, a similar issue presented itself with Unity3D itself. Every few weeks a new point release became available. Often these releases would be bug fixes and change little except the stability. Occasionally, an update would change library files, rename or remove functions and generally re-organise the code base. The solution to these issues was the same as with VRTK in that eventually a release freeze was implemented.

9.2.2 Tracking

The tracking software developed by Michael Rietzler and Florian Gierhart from the University of Ulm functioned well (GEISELHART et al., 2016; RIETZLER et al., 2016). Their kind assistance in allowing the RE to beta test their software and ultimately include it within the DAAC project proved key. As with many code bases that are under-developed, the software did not come with a set of instructions, and at least in its first beta release iterations, proved to be non-intuitive to set up. However, as the software progressed, a simple and easy to use Graphical User Interface (GUI) came to the fore, and set-up over Remote Desktop Protocol (RDP) became possible.

The actual set-up of multiple Kinects to cover a manufacturing cell was relatively simple. The time consuming aspect was ensuring that the spatial relationship between the sensors was correctly mirrored in the virtual space. To correctly align the two spaces, a new gameObject needed to be created in the VE. This object acted as an anchor, or origin point for the tracking space. With the anchor in place, aligning the tracking space to its digital twin in the VE just required rotating and translation of the anchor point. The tracking data could be imported as a live stream into the editor itself. This allowed for the avatar to animate within the editor, which meant that a user could stand at a known location within the real manufacturing cell, and then the avatar could be aligned to the virtual equivalent. Getting this exact often required a few locations and multiple re-orientations, but once set up correctly, the tracking space was accurate to within 100 millimetres. On average it would take around three to four minutes to get the correct alignment and calibration for the sensors and software. If the number of sensors were increased over five then that time would increase only by the time it took to set-up the sensor position. Setting up the alignment of the digital twin to the real manufacturing cell was more time consuming, but having a desktop built into the frame meant that the global positioning gameObject could be rapidly repositioned and re-orientated to allow for better alignment. With a some practice the process would take less than ten minutes to complete.

With the Kinect sensors mounted to the DAAC frame, it was easy to mount the corresponding brix computer. The requirement for each sensor to have its own computer, and each computer being limited to a single sensor was cumbersome, and restricted both the methods and the areas that could be tracked.

9.2.3 Work Instructions

The work instructions worked correctly, although they did require several correlating scripts to function. The first was server based, and loaded the instructions into an array, which was called by each user as they joined the server. The instructions had to be stepped through on the server to avoid the different users falling out of sync. This was less than ideal, but serves as a good example of where global variables could have been used to good effect.

The instructions were clear in both the VE and when projected out into the real manufacturing cell. However, they were severely limited. In practice, cycling through the work instructions simply enabled one set of gameObjects and disabled the rest of the array. Although functional, the method introduced several limitations:

- The game objects were static within the VE as they needed to be projected to the correct location
- The user had to cycle through them on button or key press
- The user could not add or amend the existing instructions

A different methodology could have introduced a number of advantages, some of which will be covered in Section 9.3.3. The work instructions served their purpose in the project, and proved that work guides and laser pointers could be projected from camera points within the VE into the real manufacturing cell.

9.2.4 Projection

The users within the VE had ability to pin point exact locations in the real manufacturing cell to within a few millimetres. The laser pointers proved to be one of the most useful aspects of the project, not just in communicating with users within the cell, but also in aligning the projectors. This was a task that required fine tuning almost every time the system was starting. Despite strong mountings on the frame, the projectors were very prone to falling out of alignment with each other. The frame was on wheels itself, but these could be locked out once the desired position was arrived at. The projectors then were positioned, and rotated on their triple axis mounts. The mounts however were not up to the task of holding the projector in the correct location for long periods of time. To help reduce the time it took to align the projectors, a user could set-up a group of laser pointers that would fall where the projectors overlapped. With three or more laser points shown in the row of adjacent projectors, the effort was reduced to just aligning the points.

To see the laser points clearly, the background of the VE was culled from the cameras. This meant that only the laser points and the work instructions were projected out. Even then, if the factory floor was bathed in bright sunlight, it became hard to discern some of the finer points. To help accommodate this a basic script that caused the light points and the work instruction cable guides to pulse with light was created. The variance in light made it easier to discern the points, from the objects that it shone upon, in bright light.

Although the projectors could be aligned using the laser pointers, the light itself was not correctly mapped to the objects that the pointers were shining on. A level of pseudo projection mapping was taking place because the objects in the VE mirrored those in the real manufacturing cell. There was still a noticeable discrepancy in how and where the light fell and great care was required to ensure that alignment and the visual results were satisfactory. The pseudo mapping caused by the distorting effects of the VE could have been more effective if the correct CAD of the wing skin had been imported. However, since a laser point scan was used as the base, no real surfaces existed within the VE. To try and accommodate this issue, a set of objects were created within the VE that only the laser pointers would collide with. These objects were set-up to closely mimic the geometry of the wing skin but would always fall short as they were a simulacrum of the real thing.

9.2.5 VoIP

The VoIP elements of the project were both effective, loud and clear. The decision to only enable the VoIP broadcast when the users required it, worked well. Despite using Vorbis compression, the computational load of sending live VoIP across the network proved high; with multiple users all talking at once, the communication could become weak and distorted.

The distortion also occurred at times when a lot of background noise was present. It is the RE opinion that this is to do with the codec's method of compression, and it could be significantly improved if a better high and low end filter was applied before compression. However, the software would always recover if the VoIP function was toggled again.

9.3 Future Enhancements

Although the DAAC can be considered fully functional, there are a number of areas where the project could be improved. These range from minor tweaks to using an entirely new engine. The penultimate section of this chapter will briefly discuss a number of improvements that could be applied to take the project forward.

9.3.1 Virtual Environment

The virtual environment is functional but already looks and feels dated compared with what is available within the games industry. Although it is always optimal that the VE looks as good as it can, having a state of the art environment serves another less obvious function. Today's young engineers will have grown up as digital natives and if we are to entice them and keep them working within a VE, we need to ensure we meet their expectations; expectations that have been manufactured by AAA games studios. There could be an optimal point between a cutting edge games environment, having a team of lead games designers and the reality of modern manufacturing.

Modern engines offer more than just good looks (EPIC, 2018). The Unreal 4 Engine offers better dynamic light calculation; a better quality render engine and better thread management, to name but a few advantages. The Unreal Engine is built and scripted in C++, offering a less complex game script called UScript instead of allowing JavaScript. This means that there is no requirement for the JAVA runtime environment, which reduces the computational load and requirements.

A new game engine is not the only future direction the RE would recommend. The project offers a good opportunity to explore the gamification of the workplace itself. A system such as the DAAC could easily be adapted to seek achievements that could be worked towards, such as a set of score boards not too dissimilar from the KanBan boards made famous by Toyota. Gamification could also be used to make the training aspects more fun and more engaging. Teaching workers where to put a nut or bolt repetitively could well lead to boredom and dissatisfaction, whereas adding more intricate GUI interfaces and feedback mechanisms could help alleviate that.

Following the same vein of thought as gamification would be the suggestion of the addition of an in-engine recording mechanism. At present, an external program has to be used to record elements of the project. This seems unnecessary to the RE and there are a number of methods available on the Unity3D store that allow recording ‘in-game’. This would make the recording of best practice and elements for the quality chain more useful and more robust.

Other GUI elements that should be incorporated are in-situ menu systems. The current GUI is very limited, and offers only a few options to users. More options, sub menus and more levels of interaction with the components could benefit users both in the VE and within the cell. One easy improvement would be sub-menus on objects. This would work as a menu system that can be accessed when the user is holding angameObject. For instance, when a user picks up a laser pointer, they could access a submenu that would allow them to change the colour of the light, fix the laser in place and/or spawn a new laser pointer. Equally a sub menu for the work instructions would allow the users to load different sets of work instructions and skip to sections of interest by using the menu. The menu system could even be used to load a variety of different training exercises or simulations.

9.3.2 Tracking

As mentioned in section 9.2.2 there were no unique identifiers assigned to the different real manufacturing cell users. With multiple tracked users, this could lead to avatar swapping when users left the tracking area to return later. A number of methods could be used to assign a unique identifier:

Skeletal data set - The tracked skeletal data of an individual is relatively unique. The dependency on body shape means that the Kinect skeletal data frames tend to be unique. Higher levels of accuracy could be achieved by measuring the relationship between multiple joints. It remains to be seen how many users could be handled in this way before the margin of error between users became enough to confuse the system.

RFID - Radio-Frequency Identification (RFID) tags could be used, these would need a separate tracking network from the Kinect sensors and therefore increase the number of systems embedded within the DAAC. The combination of the RFIDs and the skeletal data should be done as an extension of the Ulm Fusion tracking software. Coupling the users RFID with an avatar within the VE would require a new thread and greater computational load than parsing the RFID as part of the skeletal data.

The use of unique identifiers would be beneficial not just for ensuring the same avatar was used for each user, but also as a unique key. The unique key could be used as part of a database that could save other details about the user. These need not be performance related, instead it could information about the user's training certificates. These could be used to verify whether the user's training was up to date. It could also be used to ensure the right skill sets are present when performing certain tasks. The verification of this would add greater detail and therefore value to the quality chain. The increased level of tracking would also allow for bespoke instructions to be delivered to that user. This would allow for multiple tasks to be performed and assisted within the same cell.

The issue of storing users data about performance and optimisation is controversial. However, with a system like this it would be possible to further improve the utilisation, routing and optimisation of the human components of the manufacturing cell.

Another area that could be improved with tracking is the use of fiducial markers. Using the RGB camera feeds, the Kinect's Software Development Kit (SDK) can be used to recognise fiducial markers. This could be used in a number of ways, from tracking parts and resources within the cell, to the checking in and out of tools. Again the verification of what tools and materials were used when, would add significantly to the quality chain.

Fiducial markers could also be used to reduce the set up time; by placing them in known locations and having a gameObject assigned to them within the VE, the alignment of the tracking space and the VE could be achieved in less time.

9.3.3 Work Instructions

The work instructions within the DAAC were limited and contained only a linear path of assembly. With the sub-menu system previously suggested, this need not be the case. It would be good to integrate and import method using a common file format such as JSON that would allow for work instructions to be quickly and easily imported into the VE.

A greater range of testing is required to prove the value of using such a system as the DAAC. As such a larger range of work instructions should be coded into the VE so that the best use cases can be identified and developed further.

An improved GUI system would allow for the users to take full advantage of the nature of digital work instructions. Options to make notes as text images and video, on each set of instructions, would allow for the real world experience to be pulled back into the VE training systems. With the Kinect sensors it is even possible to capture and record the point cloud data of any process or notes a users wishes to make.

The capture process could also be used to create animated work instructions. The use of animation within a VE allows the users to see the task from any angle they wish. This could help reduce the difficulty in understanding complex processes. Similarly animations could be used to aid in the delivery of the projected instructions. These could be controlled from the work instruction sub menus, either from within the real manufacturing cell or the VE.

The work instructions could be delivered to different users depending on their location and could be controlled by their actions. In this way the work instructions would be triggered by a users location within the real manufacturing cell. This could be accomplished by using triggers within the VE that were set off when a users avatar was close to them. This could in turn be used to display more context specific information to that user.

The ability to record users' movements and actions within the game engine over a period of time could allow a users action to be easier review at a later date. This could help in self taught training, assisted learning, capture of best practice and providing more data for the quality chain.

9.3.4 Projection

The projection system is an area where a few improvements could be made, as it is dependant upon the other DAAC systems. One way the projection system could be improved would be to use smaller and more portable projectors. These could be placed at will and oriented to ensure maximum coverage of the work space. Furthermore, brighter bulbs and a method to dim the light on the factory floor would be useful. Perhaps in addition to projectors, a series of remote servo driven lasers could be used. This would introduce a significant boost in lumens where it was required.

Within the VE the projection mapping could also be improved. The pseudo projection mapping caused by the objects within the VE worked relatively well, however an understanding of occlusion and a more accurate mapping to the objects would improve the experience. One way this may be possible would be to use the depth sensor data from the Kinect sensors and map the projectors onto that.

Another area that has been touched on in the previous section 8.2 is using the user tracking to deliver targeted work instructions, which change dependant on the workers location. The projectors could project the information the users required on any object that is near enough head height, and flat enough to be readable. A further extension to this could be head tracking, where the system is smart enough to know where a user is looking.

9.3.5 VoIP

The VoIP integration into the VE meant that the package was a single binary. However the VoIP could have been a separate system entirely, as the VoIP system had no interrelations with systems. The ability to enable and disable voice communications through the application meant that users need never leave the VE. However using the Unity3D system to code and decode meant a significant increase in computational load.

A partial solution to this would be to have a separate machine that acted as a voice communications server. This would handle the receipt and transmission of encoded voice communications, effectively freeing up the VE server. To free up the VE on the client side would require integration of another program that could run in tandem with Unity3D. This would require a VoIP system that had open API hooks that could be called to enable and disable transmission. This would also reduce the network traffic that was run through the VE server.

One final improvement to the VoIP system could be voice commands. This would further free up users hands and allow them to change work instructions whilst carrying out work. Although voice recognition is now an established technology the RE is not convinced that it would be effective in a loud manufacturing environment.

9.4 User Feedback & Testing

The DAAC was tested extensively by the RE while it was in development. On completion, a set of ten volunteers from within the Advanced Manufacturing Research Centre (AMRC) and RR were used to gain further insight into possible issues and the overall usability of the system. The volunteers filled out a the user testing forms that can be found in Appendix C.3. The users were asked to complete the work instructions which are provided in Appendix C.1. The results of the tests have not been included as it was deemed that the sample size was too small to offer any insight other than the qualitative information rendered in both the survey but also through discussion with the individuals during and after the testing took place.

9.4.1 Feedback

Overall the users rated the DAAC as both easy to use and intuitive. All testers stated that it was easy to understand what was taking place within the VE and that the representation of objects within the VE was clear and concise. The most effective means of communication was the shared work instructions, although no method was deemed to be anything less than functional.

9.4.1.1 Issues or Difficulties

Some criticism was levelled at the VE and VR experience as a whole. Several users complained that the cell felt too cluttered and that being able to spawn tools from a menu system would be quicker and easier than picking them from a tool box. The size of the user prefabs were found to be overly large and overbearing.

The instructions on the VR menu system required greater clarification as to which button did what. This is in part because although the buttons appeared on keypad press, the icons were unclear.

9.4.1.2 Improvements or Changes

A number of improvements were suggested by the users. These included:

- Video chat between users
- 3D drawing in environment similar to Googles TiltBrush (Google, 2018a)
- Removal of projector users prefabs
- An increase in the number of tracked users

9.4.1.3 Where do you see the system being used?

All the volunteers identified the value in using the system to share best practise, and to allow global collaboration between project groups.

> "An excellent remote support system that would allow for companies to share manufacturing processes to gain the most effective and efficient workflow, the ability to share in real time from one site to another"

A number of other uses were suggested by the volunteers, the most common are listed below:

- Training
- Expert technical support
- Repair Maintenance and Overhaul (RMO)
- Design and on-site assembly within the construction industry
- Tourism
- Remote teaching and lecturing

9.4.1.4 Other suggestions

A common suggestion was that the DAAC was overly large and unwieldy and that a much smaller and portable version could provide the same level of benefits. Another suggestion was that a level of AR should be incorporated with the use of Microsoft's HoloLens (MICROSOFT, 2018a) or other mixed reality technology (MICROSOFT, 2018b).

9.5 Nuclear Manufacturing

Although the DAAC was set-up using a wing-skin from Boeing, it is easy to imagine that the same processes could be transferred to the assemble processes associated with pressure vessels, heat exchangers and other more nuclear industry specific components. The reason these vessels were not used was because of the associated Internet Protocol address (IP) and difficulty in procuring one for the research.
So where does the DAAC fall within the remit of nuclear manufacturing? This thesis began by highlighting a number of issues being faced by nuclear manufacturers within the UK. It is the fervent belief of the RE that, coupled with other modern manufacturing techniques, the DAAC could be developed to help address the following challenges.

9.5.1 Decommissioning older reactors

With a large section of the Advanced Gas Reactor (AGR) fleet due for decommissioning in the next decade, having the ability to train staff in a safe VE for work that will be completed with a nuclear hot zone could hold real value. Through repeated training in the VE staff can work to minimise the dose times that affect them. This will allow them to accomplish more work within the specified dose allowances.

This could be extended by the incorporation of 3D nuclear heat maps. The maps could be used to calculate the trainee's affected dose. With such a system, dose optimisation and the full body tracking users could adapt their poses to minimise dose to high risk areas. With further research and future verification it is not unfeasible to believe that such VE simulations could help to increase the certainty of worker's dose rates. The DAAC could be used here as both the VE that users train within, but also as part of the verification process to ensure that users are following the required actions within the nuclear hot zones. A comparison of the predicted doses from the VE and the actual doses taken from dose meters could then be done. In this way, it is possible that VE dose models could be certified for use.

Outside of dose calculations, the decommissioning of a Nuclear Power Plant can be seen as the reverse of what takes place in a manufacturing environment. In this regard, the existing benefits discussed within Section 9.1 would also apply. One of the major areas could be the benefits of adding more data to the quality chain. Use of the DAAC could prove that the standards of regulation were met.

Each AGR in the UK is slightly different due to the increase of experience gained in building and running the previous reactors. However the majority of the components of the AGRs are the same and, as such, recording of the decommissioning process of one reactor could prove valuable when decommissioning the next reactor. The small number of reactors and the large period time between their decommissioning processes, means that capturing the experiences and lessons learnt on one would be valuable for the next. This would be especially true since the length of time it takes to decommission a reactor could mean a new work force is employed between reactor decommissions.

Another area where the DAAC could prove effective would be in the importing of scanned data from legacy components or areas of nuclear plant. Some older plants and manufacturing facilities were designed on paper or 2D CAD. The ability to import point cloud scanned data, would allow for these facilities and legacy parts to be imported into the VE. Thus a digital twin could be created for parts that previously had little to no digital data. Work processes could then be designed and trialled using the digital twin rather than the legacy parts. Previously this would have to be done on paper, and using scaled models. The use of the DAAC would not only allow for one to one size representation of the parts, but would allow for training and then remote assistance to be used when performing any planned work on the parts or facilities.

9.5.2 Providing technical assistance and safety verification

There is a very high level of safety regulation within the nuclear sector in the UK. When coupled with the slow rate of growth within the UKs nuclear sector, this has led to a low number of experts within each field.

With technologies such as the DAAC, companies would have the ability to connect experts to remote locations with relative ease. This would allow for a central repository of expert knowledge to be linked with users across the sector. The ability to provide remote assistance from industry experts, would help train new employees as well as ensure that the high standards required in the industry were met.

The use of the VE could also be used in training new Office of Nuclear Regulation (ONR) inspectors in plant inspection and safety verification. The diversity of AGR designs, the potential for multiple new reactor designs and the complexity of Gen 3+ reactors, the ability to rehearse and practise inspection techniques and methods within a VE could prove valuable. With the DAAC, new inspectors could call on the help of existing inspectors for assistance on location. The use of a digital twin of the intended inspection area would allow the inspectors to rapidly identify any divergent features; this could then be saved for further inspection within the VE.

9.5.3 Manufacture of the new nuclear build Gen 3+ or SMRs

The manufacture of components for the Gen 3+ and Small Modular Reactors (SMRs) reactors will be high value and low volume. This will necessitate that manufacturing facilities will be:

High conformance to codes such as ASME
Such codes and requirements can be baked into the delivered work instructions, meaning that staff are aware of the correct regulations while performing a corresponding operation.

Highly Trained Staff
The DAAC will offer the ability to train staff inside and outside of the manufacturing cell. Furthermore it can provide staff with examples of best practice as well as a direct link to the company's experts. The actions of staff can be monitored and kept for benchmarking and as part of continuous improvement cycle.

Reduction of Manufacturing Errors
Getting manufacturing processes right the first time, could be improved by the users gaining assistance from remote parties. Also the projection of work instructions into the manufacturing cell could help to reduce the likelihood of a process being carried out incorrectly. With special or codified processes, the ability to transmit work instructions both aurally and visually to the work, onto the components they are working on could help to reduce user error, as well as ensure that correct methodology is observed. Furthermore the additional potential of oversight from the VE users could act in a similar fashion to Foucaults' Panopticon. Where the mere idea of being observed at all times was thought to induce better behaviour and conformance from prisoners.

Flexible facilities and staff
The ability to cross skill staff will allow for higher utilisation rates within the factory. The DAAC can also be moved around a factory floor on its coasters, to be re-aligned, as a new manufacturing cell is created. The VE and digital twins within the VE can be reconfigured as frequently as desired to allow for maximum flexibility.

Diverse range of components
A large range of components will leave little room for learning through multiple build iterations. The ability to train in a one to one scale environment could help users prepare for new processes, or help reacquaint users with ones not performed for some time. The DAAC also allows for remote

assistance, so that any challenges that manufacturing cell users are unsure about can be checked with more experienced or qualified users within the VE. ASME manufacturing codes do not allow for approval of processes from one manufacturing facility to another, however the use of practises to train individuals across the value chain is still pertinent. It could also help to accelerate the qualifications of one facility by training the staff in the methods of another. Which in turn could assist in gaining the correct ASME qualifications for that site.

Specialist equipment for handling & manufacture
The DAAC cell could be used to train multiple staff in the requirements for performing complex operations. Such as the handling and lifting of very large components. The VE allows for as many users as required to perform the operations correctly. The training for use of specialist equipment could also be performed within the VE and then checked and verified using the DAAC. This would lead to ensuring that users were competent at performing the operation before attempting to do so.

Slow uptake of innovative techniques
The DAAC can be used as an effective test bed for new techniques and processes. It offers both a fully VE adaptive experience, alongside the possibility of verification within the real.

Documentation
Through the use of digital work instructions and various feedback mechanisms the DAAC can offer an updated and more versatile method for adapting documentation. Furthermore more context is saved when using the DAAC to record and store data of operations. This will allow for a better interpretation of documentation in the future.

Long lead times
The ability to track the human components within a manufacturing process will help add more assurance to the actual lead time for component manufacture. Through the sharing of best practise across manufacturing cells and factories, the understanding of what goes into creating a product lead time will increase, and overall lead times can be reduced.

9.5.4 The skills gap

The DAAC was designed to help capture both good practise and help disseminate expert knowledge through a manufacturing chain. With an ever increasing skills gap in the nuclear sector, between retiring and newly recruited staff, this could prove vital in both the storage and transferral of skills and information. Work flow and manufacturing processes by experts within the field could be captured in one location, and shared not just across geographic distances between manufacturing facilities, but also across time for future generations to learn and master.

9.5.5 Public support for the nuclear industry as a whole

This is one challenge that the DAAC fails to properly address. It may be true that the use of modern technologies will help re-assure a very wary public. However this is outside of the remit of this thesis, and could possibly be a topic of its own.

CHAPTER 10

Conclusion

> Modern manufacturing will be conducted in structured, modular and smart factories. These will use Cyber Physical Systems (CPSs) in value adding processes, whilst recording the data to the Digital Twins of the product. The CPSs will communicate over the Internet of Things (IoT) network within the confines of; the smart factory; the company as a whole; and with people, all in real time. *Chapter 5*

Research on the nuclear industry identified a skills gap between older workers retiring and new employees recruited for the nuclear renaissance. A severe reduction in manufacturing for nuclear plants has led to poor passage of manufacturing knowledge between these generations. The use of Mixed Reality (MR) has always been costly, and driven by business sectors that have enough revenue to fund it. These have traditionally been the military, medicine, universities, and the automotive industry. Research on integrating MR with manufacturing has been limited to the pre-manufacture product life cycle and with limited scope. The start of the 4^{th} Industrial Revolution (IR4.0) is driven by the use of interconnected systems that use rapid data transfer to allow for greater optimisation within manufacturing. However despite the creation of augmented operators there is a lack of data being generated by or about the operators on the manufacturing floor themselves.

The Digitally Assisted Assembly Cell (DAAC) is, in effect, a Digital Twin of a manufacturing cell and its users (Figure 10.1). It is able to communicate over a network, within the confines of a smart factory, the company as a whole and with people, all in real time. It uses Virtual Reality (VR) and in-situ projection of work instructions to provide context-sensitive visualisation of data. The DAAC can also be used for training and “upgrading the workforce”(RÜSSMANN et al., 2015), which will aid in the adoption and adaptation of modern manufacturing techniques associated with the IR4.0. This could prove particularly useful within the nuclear industry in closing the skill gap from the side of the younger generations. The DAAC could also serve to help preserve the knowledge of the older generations of manufacturers. By recording manufacturing processes, work instructions, and best practise of an ageing work force for future generations to review and use as a basis of training and improvement. It also assists in a manner that is intuitive to non digital natives and digital natives alike. It allows for anyone within a company or value chain to connect with shop floor workers. The DAAC uses low cost, readily available materials and technology which further reduces the barrier to entry for Small to Medium Enterprises (SMEs). The project could also be scalable in size to fit different levels of production.

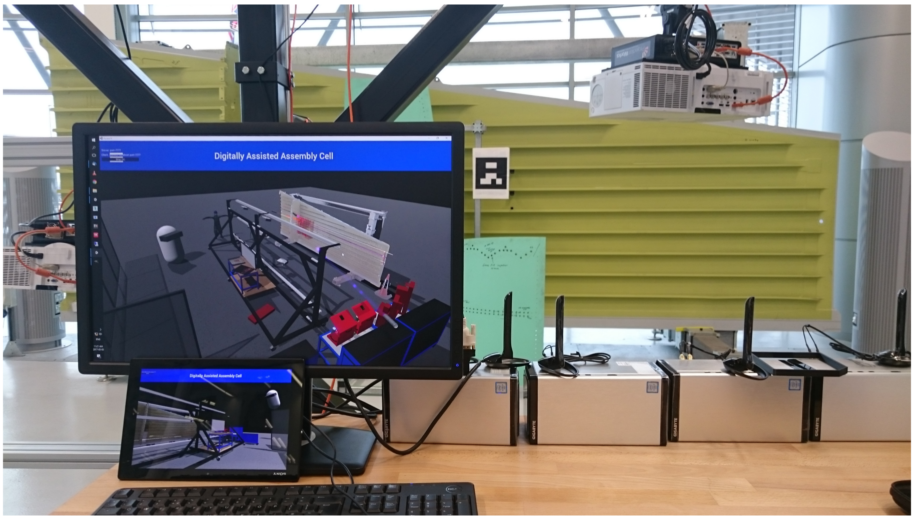


Figure 10.1: The Android and desktop client mounted within the real DAAC manufacturing cell

It is for these reasons that the Research Engineer (RE) believes that the DAAC is a good candidate for a CPS that addresses the challenge of the “Double Digital Divide” (ACATECH, 2016a).

The RE concludes that the DAAC is a Cyber Physical System (CPS) that both:

- Accounts for the interaction of interdependent behaviours of cyber & physical components to achieve system level functionality
- Conjoins continuous dynamics of the physical components and discrete dynamics of cyber components

However the RE would stop short of saying the DAAC was realising the full potential of CPS as defined by Lee et al.

> “To realise the full potential of CPS we will have to re-build computing and network abstractions. These abstractions will have to embrace physical dynamics and computation in a unified way”(E. A. LEE, 2008)

The abstractions created in this project do embrace physical dynamics but the computational approach is not unified. The DAAC is a fragmented set of sub-systems that feed into a single Virtual Environment (VE), where the main form of interaction is visual. There is no co-dependence of systems and one system does not drive another. Instead the DAAC represents more of a pseudo -CPS of interrelated systems. The potential for co-dependence and internal reliance exist and event driven processes could be created within the VE but would be a possible future extension of DAAC.

The DAAC serves as a proof of concept and demonstrates the possibilities of using a multi-user, multi-platform VE as a digital twin to a real manufacturing cell. Furthermore, it demonstrates that a variety of methods of bi-directional links can be created to further enhance the value of mixed mode manufacturing techniques. It is the belief of the RE that the use of MR could reduce cost and overall lead times within the civil nuclear manufacturing industry.

10.1 Future Research

The RE has identified a number of ways that the project could be improved in Section 9.3. Below the RE lists a few recommendations for future academic research surrounding the DAAC concept:

Rigorous performance and application testing The current iteration of the DAAC has not been benchmarked, or undergone testing outside of a small user base. This means that there is need and scope for wider user testing and iterative redesign to improve the system. Once the system has been improved it could be used to start verifying how effective it is in addressing the issues it was created for. A larger sample size of users would be required to gain any confidence in the statistical results of user testing. It could then be used to verify other theories surrounding the use of user tracking within smart factories, CPSs and other IR4.0 systems.

Variation in work instructions and scenarios A greater diversity of work instructions should be imported into the VE so that the effectiveness of the system can be tested against as many manufacturing processes as possible. More bespoke nuclear manufacturing processes should be tested on large sample sizes of users to ensure statistical relevance and provide results varied by manufacturing technique. It is possible that the DAAC will be more effective with some manufacturing processes than others.

Investigation into alternative game engines The unity3D engine proved to be suitable for the initial creation of the DAAC, but it is possible that other game engines could provide improvements. The Unreal 4 Engine can be built from source, which would allow for optimisations and improvements to be made at the root of the code base. This could substantially reduce the number of separate processes running, and their variation. It could also improve performance by having the thread management done solely by a single governor. It is the REs opinion that the DAAC should be re-written from the ground up if it is to reach its potential.

Test the limitations of the DAAC system This would improve increasing the tracking space as far as possible, to find out how large an area could be tracked by a single system. This would help verify the optimal number of tracking sensors to use within a given cell size. Furthermore work on if and how many DAAC systems could be interlinked. This would dictate how large a cell could be digitally twinned. The size of the virtual environment should also be correspondingly tested.

It would also be of great use to test integration of the DAAC with other data streams or CPSs. The real time import and use of other data streams could reap large benefits and prove to be foundational for integrating the DAAC into larger systems.

Creation of a more compact DAAC for greater portability Although the limitations of the

DAAC tracking systems have not been investigated, the RE believes that a smaller more portable version could prove useful for in-service training and operation. A smaller system could be moved onto sites of both nuclear power plants and manufacturing facilities.

The RE would also like to recognise that the above list is not exhaustive, and that the scope for research on such a system is as wide as ones imagination.

Bibliography

ABDULLAH, T.A, K. POPPLEWELL and C. J PAGE: 'A Review of the Support Tools for the Process of Assembly Method Selection and Assembly Planning'. *International Journal of Production* (2003), vol. 41(11): pp. 2391–2410 (cit. on p. 40).

ABELE, EBERHARD, ARNO WÖRN, JÜRGEN FLEISCHER, JAN WIESER, PATRICK MARTIN and ROBERT KLÖPPER: 'Mechanical module interfaces for reconfigurable machine tools'. *Production Engineering* (2007), vol. 1(4): pp. 421–428 (cit. on p. 92).

ACATECH: *Kompetenzen für Industrie 4.0 – Qualifizierungsbedarfe und Lösungsansätze*. [White Paper]. [Accessed 23 October 2017]. Munich, Germany: Deutsche Akademie der Technikwissenschaften, 2016. URL: http://www.acatech.de/de/publikationen/stellungnahmen/acatech/detail/artikel/kompetenzen-fuer-industrie-40-qualifizierungsbedarfe-und-loesungsansaetze-1.html (cit. on pp. 79, 80, 84, 96, 97, 182).

– *Recommendations for implementing the strategic initiative INDUSTRIE 4.0*. [White Paper]. [Accessed 23 October 2017]. Munich, Germany: Deutsche Akademie der Technikwissenschaften, 2016. URL: http://www.acatech.de/fileadmin/user%5C_upload/Baumstruktur%5C_nach%5C_Website/Acatech/root/de/Material%5C_fuer%5C_Sonderseiten/Industrie%5C_4.0/Final%5C_report%5C_%5C_Industrie%5C_4.0%5C_accessible.pdf (cit. on pp. 83, 84, 87, 88, 96).

ACKOFF, RUSSELL L: 'From data to wisdom'. *Journal of applied systems analysis* (1989), vol. 16: pp. 3–9 (cit. on p. 33).

ACTON, THOMAS, RAIJA HALONEN, KIERAN CONBOY and WILLIAM GOLDEN: 'DeLone & McLean success model as a descriptive tool in evaluating the use of a virtual learning environment'. (2009), vol. (cit. on p. 94).

AIDEN LOCKWOOD Robin Scott, RICHARD LANYON-HOGG: *ADG2994 Development of a 4IR (Industry 4.0) Ready Machining Facility*. Presentation. [Accessed 23 February 2017]. Sheffield, UK: AMRC Integrated Manufacturing Group, 2017 (cit. on pp. 81, 87, 88, 92).

AIRBUS: *A350 XWB from Virtual Reality to Reality*. [Accessed 02 November 2015]. 2013. URL: http://videos.airbus.com/video/03682cfa0bbs.html (cit. on p. 75).

AKPAN, JUSTICE I and ROGER J BROOKS: 'Experimental investigation of the impacts of virtual reality on discrete-event simulation'. *Simulation Conference, 2005 Proceedings of the Winter*. IEEE. 2005: 8–pp (cit. on p. 61).

ALKAYA, EMRAH, MERVE BOGURCU, FERDA ULUTAS and GÖKSEL NIYAZI DEMIRER: 'Adaptation to climate change in industry: improving resource efficiency through sustainable production applications'. *Water Environment Research* (2015), vol. 87(1): pp. 14–25 (cit. on p. 96).

ALLENDORF, JDF, M BESSLER, RL WHELAN, M TROKEL, DA LAIRD, MB TERRY and MR TREAT: 'Postoperative immune function varies inversely with the degree of surgical trauma in a murine model'. *Surgical endoscopy* (1997), vol. 11(5): pp. 427–430 (cit. on p. 51).

ALLISON, DON, BRIAN WILLS, DOUG BOWMAN, JEAN WINEMAN and LARRY F HODGES: 'The virtual reality gorilla exhibit'. *Computer Graphics and Applications, IEEE* (1997), vol. 17(6): pp. 30–38 (cit. on p. 50).

AMRC: *Soluis Group Secures £1 million Innovate UK grant for the construction industry*. [Accessed 02 November 2017]. 2017. URL: `http://www.amrc.co.uk/news/soluis-group-secures-1-million-innovate-uk-grant-for-the-construction-industry` (cit. on p. 58).

– *University of Sheffield AMRC assist DESTACO in navigating the world of virtual reality*. [Accessed 02 November 2017]. 2017. URL: `http://www.amrc.co.uk/news/university-of-sheffield-amrc-assist-destaco-in-navigating-the-world-of-virtual-reality/` (cit. on p. 58).

ANDERSON, ABRAHAM and ZHIPING WENG: 'VRDD: applying virtual reality visualization to protein docking and design'. *Journal of Molecular Graphics and Modelling* (1999), vol. 17(3): pp. 180–186 (cit. on p. 52).

ANDERSON, NATE: *Confirmed: US and Israel created Stuxnet, lost control of it*. [Online]. [Accessed 23 October 2017]. 2012. URL: `https://arstechnica.com/tech-policy/2012/06/confirmed-us-israel-created-stuxnet-lost-control-of-it/` (cit. on p. 88).

ANDRESEN, PETER L and F PETER FORD: 'Life prediction by mechanistic modeling and system monitoring of environmental cracking of iron and nickel alloys in aqueous systems'. *Materials Science and Engineering: A* (1988), vol. 103(1): pp. 167–184 (cit. on p. 18).

ANTONYA, CSABA and DORU TALABA: 'Design evaluation and modification of mechanical systems in virtual environments'. *Virtual Reality* (2007), vol. 11(4): pp. 275–285 (cit. on p. 59).

ARAS, RIFAT, YUZHONG SHEN and AHMED NOOR: 'Quantitative assessment of the effectiveness of using display techniques with a haptic device for manipulating 3D objects in virtual environments'. *Advances in Engineering Software* (2014), vol. 76(0): pp. 43–47 (cit. on p. 45).

AREA: *Digitally Assisted Assembly at Factory 2050*. [Accessed 02 November 2017]. 2017. URL: `http://thearea.org/digitally-assisted-manufacturing-factory-2050/` (cit. on p. 58).

ARENA: *Manufacturing Simulation Software*. [Accessed 30 September 2014]. Rockwell Automation. 2014. URL: `https://www.arenasimulation.com/industry-solutions/manufacturing-simulation-software` (cit. on p. 63).

ARMS, L, DIANNE COOK and CAROLINA CRUZ-NEIRA: 'The benefits of statistical visualization in an immersive environment'. *Virtual Reality, 1999. Proceedings., IEEE*. IEEE. 1999: pp. 88–95 (cit. on p. 51).

ASME: *Boiler and Pressure Vessel Code 2013 Edition*. [Online]. [Accessed 30 March 2013]. 2013. URL: http://www.asme.org/kb/standards/bpvc-resources/boiler-and-pressure-vessel-code--2013-edition (cit. on pp. 16, 18).

AUTODESK: *AutoDesk 3DSmax*. [Accessed 02 November 2017]. 2017. URL: https://www.autodesk.com/products/3ds-max/overview (cit. on p. 106).

– *AutoDesk StingRay*. [Accessed 02 November 2017]. 2017. URL: https://www.autodesk.com/products/stingray/overview (cit. on p. 58).

AUTODESK: *Autodesk - 3D Design, Engineering & Entertainment Software*. [Accessed 17 April 2018]. 2018. URL: https://www.autodesk.co.uk/ (cit. on p. 138).

AZEVEDO, AMÉRICO and ANTÓNIO ALMEIDA: 'Factory templates for digital factories framework'. *Robotics and Computer-Integrated Manufacturing* (2011), vol. 27(4): pp. 755–771 (cit. on p. 92).

BAHETI, R. and H. GILL: *Cyber-physical systems*. Impact Control Technol, 2011: pp. 1–6 (cit. on p. 84).

BAHETI, RADHAKISAN and HELEN GILL: 'Cyber-physical systems'. *The impact of control technology* (2011), vol. 12: pp. 161–166 (cit. on pp. 84, 87, 94).

BAL, MERT and MAJID HASHEMIPOUR: 'Virtual factory approach for implementation of holonic control in industrial applications: A case study in die-casting industry'. *Robotics and Computer-Integrated Manufacturing* (2009), vol. 25(3): pp. 570–581 (cit. on p. 62).

BALDWIN, CARLISS YOUNG and KIM B CLARK: *Design rules: The power of modularity*. Vol. 1. MIT press, 2000 (cit. on p. 82).

BALLANTYNE, GARTH H: 'Robotic surgery, telerobotic surgery, telepresence, and telementoring'. *Surgical Endoscopy and Other Interventional Techniques* (2002), vol. 16(10): pp. 1389–1402 (cit. on p. 51).

BALTSAVIAS, EMMANUEL P.: 'A comparison between photogrammetry and laser scanning'. *{ISPRS} Journal of Photogrammetry and Remote Sensing* (1999), vol. 54(2–3): pp. 83–94 (cit. on p. 61).

BARBIERI, L, F BRUNO, F CARUSO and M MUZZUPAPPA: 'Innovative integration techniques between Virtual Reality systems and CAx tools'. *The International Journal of Advanced Manufacturing Technology* (2008), vol. 38(11-12): pp. 1085–1097 (cit. on pp. 57, 65).

BATTER, JAMES J and FREDERICK PHILLIPS BROOKS: *GROPE-1: A computer display to the sense of feel*. North-Holland Publishing Company, 1972 (cit. on pp. 38, 48).

BAYLISS, JESSICA D and DANA H BALLARD: 'A virtual reality testbed for brain-computer interface research'. *Rehabilitation Engineering, IEEE Transactions on* (2000), vol. 8(2): pp. 188–190 (cit. on p. 52).

BBC: *Virtual reality: Lessons from the past for Oculus Rift*. [Online]. [Accessed 18 September 2014]. 2013. URL: http://www.bbc.co.uk/news/technology-23877695 (cit. on p. 49).

BELLINGER, GENE, DURVAL CASTRO and ANTHONY MILLS: 'Data, information, knowledge, and wisdom'. (2004), vol. [Accessed 26 Sep. 2014]. URL: http://www.%20systems-thinking.%20org/dikw/dikw.htm (cit. on p. 34).

BENALI-KHOUDJA, MOHAMED, MOUSTAPHA HAFEZ, J-M ALEXANDRE, JAMIL BENACHOUR and ABDERRAHMANE KHEDDAR: 'Thermal feedback model for virtual reality'. *Micromechatronics and Human Science, 2003. MHS 2003. Proceedings of 2003 International Symposium on*. IEEE. 2003: pp. 153–158 (cit. on p. 39).

BENFORD, STEVE, DAVE SNOWDON, CHRIS GREENHALGH, ROB INGRAM, IAN KNOX and CHRIS BROWN: 'VR-VIBE: A Virtual Environment for Co-operative Information Retrieval'. *Computer Graphics Forum*. Vol. 14. Wiley Online Library. 1995: pp. 349–360 (cit. on p. 52).

BERGBAUER, JÖRG: 'Entwicklung eines Systems zur interaktiven Simulation von Produktionssystemen in einer virtuellen Umgebung'. PhD. Germany: University Clausthal, 2002 (cit. on p. 63).

BRITISH GEOLOGICAL SURVEY (BGS): *British Geological Survey 3D visulisations*. [Accessed 02 November 2015]. 2014. URL: http://www.bgs.ac.uk/research/environmentalModelling/3dVisualisation.html (cit. on p. 76).

BICKERSTAFF, KAREN, IRENE LORENZONI, NICK F PIDGEON, WOUTER POORTINGA and PETER SIMMONS: 'Reframing nuclear power in the UK energy debate: nuclear power, climate change mitigation and radioactive waste'. *Public understanding of science* (2008), vol. 17(2): pp. 145–169 (cit. on p. 9).

BIMPLUS: *Balfour Beatty "hackers" teams up with VR experts to create H&S simulator*. [Accessed 02 November 2015]. 2014. URL: http://www.bimplus.co.uk/technology/balfour-beatty-hackers-teams-vr-experts-create-hs-/ (cit. on p. 75).

BISIM: *Virtual Battlespace 3*. [Accessed 01 October 2014]. 2014. URL: https://bisimulations.com/ (cit. on p. 55).

BLANCHARD, CHUCK, SCOTT BURGESS, YOUNG HARVILL, JARON LANIER, ANN LASKO, MARK OBERMAN and MIKE TEITEL: 'Reality built for two: a virtual reality tool'. *ACM SIGGRAPH Computer Graphics*. Vol. 24. ACM. 1990: pp. 35–36 (cit. on p. 49).

BLOOMBERG: *Rolls-Royce Drone Ships Challenge $375 Billion Industry: Freight*. [Accessed 02 November 2015]. 2014. URL: http://www.bloomberg.com/news/articles/2014-02-25/rolls-royce-drone-ships-challenge-375-billion-industry-freight (cit. on p. 75).

BMBF: *Digitale Wirtschaft und Gesellschaft - Industrie 4.0*. [White Paper]. [Accessed 23 October 2017]. Bonn, Germany: Bundesministerium für Bildung und Forschung, 2017. URL: https://www.bmbf.de/de/zukunftsprojekt-industrie-4-0-848.html (cit. on p. 80).

BOBROW, JAMES E: 'Optimal robot plant planning using the minimum-time criterion'. *Robotics and Automation, IEEE Journal of* (1988), vol. 4(4): pp. 443–450 (cit. on p. 66).

BOEHELER, W: 'Comparison of 3D laser scanning and other 3D measurement techniques'. *Recording, Modeling and Visualization of Cultural Heritage, London, Taylor and Francis* (2005), vol.: pp. 89–100 (cit. on p. 61).

BOOTHROYD, GEOFFREY: 'Product design for manufacture and assembly'. *Computer-Aided Design* (1994), vol. 26(7): pp. 505–520 (cit. on p. 63).

BORST, WILLEM NICO and WN BORST: 'Construction of engineering ontologies for knowledge sharing and reuse'. (1997), vol. (cit. on p. 89).

BOSCH: *Industry 4.0 – Germany takes first steps toward the next industrial revolution*. [Press Release]. [Accessed 23 October 2017]. 2012. URL: https://blog.bosch-si.com/categories/manufacturing/2012/10/industry-4-0-germany-takes-first-steps-toward-the-next-industrial-revolution/ (cit. on p. 79).

BOSCHREXROTH: *Bosch plant: how to put people at the heart of Industry 4.0*. [Press Release]. [Accessed 23 October 2017]. 2015. URL: http://blog.bosch-si.com/categories/manufacturing/2015/06/people-are-the-heart-of-industrie-4-0/ (cit. on p. 81).

– *Industry 4.0 Award for Bosch Rexroth Assembly Line*. [Press Release]. [Accessed 23 October 2017]. 2014. URL: https://www.boschrexroth.com/en/xc/company/press/index2-14464 (cit. on pp. 80, 81).

BOWLAND, N.W, J.X GAO and R SHARMA: 'A PDM- and CAD-integrated assembly modelling environment for manufacturing planning'. *Journal of Materials Processing Technology* (2003), vol. 138(1–3). {IMCC2000}: pp. 82–88 (cit. on pp. 58, 64).

BOWMAN, DOUG, DONALD JOHNSON and LARRY HODGES: 'Testbed evaluation of virtual environment interaction techniques'. *Presence* (2001), vol. 10(1): pp. 75–95 (cit. on p. 51).

BOWYER, ADRIAN, GARY BAYLISS, RICHARD TAYLOR and PHILIP WILLIS: 'A virtual factory'. *International Journal of shape modeling* (1996), vol. 2: pp. 215–226 (cit. on p. 60).

BRANDHERM, BORIS and ALEXANDER KRONER: 'Digital product memories and product life cycle'. *Intelligent Environments (IE), 2011 7th International Conference on*. IEEE. 2011: pp. 374–377 (cit. on pp. 87, 91).

BRECHER, CHRISTIAN, STEFAN KOZIELSKI and LUTZ SCHAPP: 'Integrative Produktionstechnik für Hochlohnländer'. *Wertschöpfung und Beschäftigung in Deutschland* (2011), vol.: pp. 47–70 (cit. on p. 84).

BRETTEL, MALTE, NIKLAS FRIEDERICHSEN, MICHAEL KELLER and MARIUS ROSENBERG: 'How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective'. *International Journal of Mechanical, Industrial Science and Engineering* (2014), vol. 8(1): pp. 37–44 (cit. on pp. 82, 97).

BROOKLYN-LIBRARY, LESLIE -: *The Drivotrainer*. [Online]. [Accessed 03 September 2014]. 2009. URL: https://www.bklynlibrary.org/blog/2009/04/21/drivotrainer (cit. on p. 46).

BROOKS JR, FREDERICK P: 'Walkthrough—a dynamic graphics system for simulating virtual buildings'. *Proceedings of the 1986 workshop on Interactive 3D graphics*. ACM. 1987: pp. 9–21 (cit. on pp. 50, 58).

BROOKS JR, FREDERICK P, MING OUH-YOUNG, JAMES J BATTER and P JEROME KILPATRICK: 'Project GROPEHaptic displays for scientific visualization'. *ACM SIGGraph computer graphics*. Vol. 24. ACM. 1990: pp. 177–185 (cit. on p. 48).

BROSZE, TOBIAS, THOMAS NOVOSZEL and HENRIK WIENHOLDT: 'High Resolution Supply Chain Management'. *PPS MANAGEMENT* (2007), vol. 12(3): pp. 55–58 (cit. on p. 88).

BROUGH, JOHN E., MAXIM SCHWARTZ, SATYANDRAK. GUPTA, DAVINDERK. ANAND, ROBERT KAVETSKY and RALPH PETTERSEN: 'Towards the development of a virtual environment-based training system for mechanical assembly operations'. *Virtual Reality* (2007), vol. 11(4): pp. 189–206 (cit. on p. 66).

BROWN, EMILY and PAUL CAIRNS: 'A grounded investigation of game immersion'. *CHI'04 extended abstracts on Human factors in computing systems*. ACM. 2004: pp. 1297–1300 (cit. on p. 29).

BRYSON, STEVE: 'The Virtual Wind Tunnel'. *SIGGRAPH'93 Course*. 1 43. 1993: pp. 2.1–2.10 (cit. on p. 49).

BRYSON, STEVE and CREON LEVIT: 'The virtual wind tunnel'. *IEEE Computer graphics and Applications* (1992), vol. 12(4): pp. 25–34 (cit. on p. 52).

BULLINGER, HJ, M RICHTER and KA SEIDEL: 'Virtual assembly planning'. *Human Factors and Ergonomics in Manufacturing* (2000), vol. 10(3): pp. 331–341 (cit. on p. 64).

BURDEA, GRIGORE and PHILIPPE COIFFET: 'Virtual reality technology'. *Presence: Teleoperators and virtual environments* (2003), vol. 12(6): pp. 663–664 (cit. on p. 57).

CARLSSON, CHRISTER and OLOF HAGSAND: 'DIVE—A platform for multi-user virtual environments'. *Computers & graphics* (1993), vol. 17(6): pp. 663–669 (cit. on p. 50).

CATER, JOHN P: 'The nose have it'. *Presence: Teleoperators and Virtual Environments* (1992), vol. 1(4): pp. 493–494 (cit. on p. 44).

CAUDELL, THOMAS P and DAVID W MIZELL: 'Augmented reality: An application of heads-up display technology to manual manufacturing processes'. *System Sciences, 1992. Proceedings of the Twenty-Fifth Hawaii International Conference on*. Vol. 2. IEEE. 1992: pp. 659–669 (cit. on pp. 51, 52).

CGN EDF-Energy, BRADWELLB: *UK HPR1000 nuclear technology design successfully moves into next phase of UK approval process*. [Accessed 07 December 2018]. 2018. URL: https://bradwellb.co.uk/gda-next-step/ (cit. on p. 11).

CHAFFEY, DAVE and GARETH WHITE: *Business information management: improving performance using information systems*. Pearson Education, 2010 (cit. on p. 35).

CHAN, CHIU-SHUI, LEWIS HILL and CAROLINA CRUZ-NEIRA: 'Is it possible to design in full scale'. *A CAD Tool in a Synthetic Environment CAADRIA '99* (1999), vol.: pp. 43–52 (cit. on pp. 51, 58).

CHAUHAN, ADITYA: *GTX 680 Maya viewport 2.0 12 million polygons Test*. [Online]. [Accessed 26 September 2014]. 2012. URL: https://www.youtube.com/watch?v=hBTpwFpbZ7Y (cit. on p. 50).

CHEN, FENG, PAN DENG, JIAFU WAN, DAQIANG ZHANG, ATHANASIOS V VASILAKOS and XIAOHUI RONG: 'Data mining for the internet of things: literature review and challenges'. *International Journal of Distributed Sensor Networks* (2015), vol. (cit. on p. 92).

CHEN, MIN, SHIWEN MAO and YUNHAO LIU: 'Big data: A survey'. *Mobile Networks and Applications* (2014), vol. 19(2): pp. 171–209 (cit. on p. 92).

CHEN, SHENCHANG ERIC: 'Quicktime VR: An image-based approach to virtual environment navigation'. *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*. ACM. 1995: pp. 29–38 (cit. on p. 50).

CHIEN, CHEN-FU and REN-TSUN KUO: 'Beyond make-or-buy: cross-company short-term capacity backup in semiconductor industry ecosystem'. *Flexible Services and Manufacturing Journal* (2013), vol. 25(3): pp. 310–342 (cit. on p. 82).

CHRISTIAND and JUNGWON YOON: 'Optimal assembly path planning algorithm for aircraft part maintenance'. *Control, Automation and Systems, 2007. ICCAS '07. International Conference on*. Oct. 2007: pp. 2190–2194 (cit. on pp. 66, 70, 75).

CHRISTOPHER, MARTIN: 'The agile supply chain: competing in volatile markets'. *Industrial marketing management* (2000), vol. 29(1): pp. 37–44 (cit. on p. 82).

CHRYSSOLOURIS, GEORGE, DIMITRIS MAVRIKIOS, DIMITRIS FRAGOS, VASSILIKI KARABATSOU and KOSMAS ALEXOPOULOS: 'A Hybrid Approach to the Verification and Analysis of Assembly and Maintenance Processes Using Virtual Reality and

Digital Mannequin Technologies'. *Virtual and Augmented Reality Applications in Manufacturing*. Ed. by ONG, S.K. and A.Y.C. NEE. Springer London, 2004: pp. 97–110 (cit. on p. 66).

CHU, CHI-CHENG P, TUSHAR H DANI and RAJIT GADH: 'Multi-sensory user interface for a virtual-reality-based computeraided design system'. *Computer-Aided Design* (1997), vol. 29(10): pp. 709–725 (cit. on p. 45).

CLEVER-MACHINERY: *Pushing the boundaries of reality helps manufacturers gain an edge*. [Accessed 02 November 2017]. 2017. URL: https://clevermachinery.com/manufacturing/pushing-boundaries-reality-helps-manufacturers-gain-edge/ (cit. on p. 58).

COLERAINE, LORD: 'Cm.2335 "THE SECOND NUCLEAR POWER PROGRAMME"'. *House of Lords Proceedings*. Vol. 258. [Accessed 04 September 2014]. London: Hansard, July 1964: pp. 894–904. URL: http://hansard.millbanksystems.com/lords/1964/jun/10/the-second-nuclear-power-programme (cit. on p. 8).

COMEAU, CHARLES P and JS BRYAN: 'Headsight television system provides remote surveillance'. *Electronics* (Nov. 1961), vol. 10: pp. 86–90 (cit. on p. 46).

CONN, COCO, JARON LANIER, MARGARET MINSKY, SCOTT FISHER and ALISON DRUIN: 'Virtual environments and interactivity: Windows to the future'. *ACM SIGGRAPH Computer Graphics*. Vol. 23. ACM. 1989: pp. 7–18 (cit. on p. 24).

CONSORTIUM, INDUSTRIAL INTERNET: *about us*. [Online]. [Accessed 23 October 2017]. 2017. URL: http://www.iiconsortium.org/about-us.htm (cit. on p. 96).

COOPERS, PRICE WATERHOUSE: *Industry 4.0: Building the digital enterprise*. [Global Survey]. [Accessed 23 October 2017]. 2016. URL: https://www.pwc.com/gx/en/industries/industries-4.0/landing-page/industry-4.0-building-your-digital-enterprise-april-2016.pdf (cit. on pp. 80, 83, 88, 96).

CRAIG, ALAN B, WILLIAM R SHERMAN and JEFFREY D WILL: *Developing virtual reality applications: Foundations of effective design*. Morgan Kaufmann, 2009 (cit. on p. 25).

CRUZ-NEIRA, CAROLINA: 'Making virtual reality useful: a report on immersive applications at Iowa State University'. *Future Generation Computer Systems* (1998), vol. 14(3): pp. 147–155 (cit. on p. 56).

CRUZ-NEIRA, CAROLINA: *VR Juggler*. [Accessed 30 September 2014]. Priority 5 Holdings, Inc. 2013. URL: https://code.google.com/p/vrjuggler/ (cit. on pp. 63, 71).

CRUZ-NEIRA, CAROLINA, DANIEL J SANDIN and THOMAS A DEFANTI: 'Surround-screen projection-based virtual reality: the design and implementation of the CAVE'. *Proceedings of the 20th annual conference on Computer graphics and interactive techniques*. ACM. 1993: pp. 135–142 (cit. on p. 51).

CRUZ-NEIRA, CAROLINA, DANIEL J SANDIN, THOMAS A DEFANTI, ROBERT V KENYON and JOHN C HART: 'The CAVE: audio visual experience automatic virtual environment'. *Communications of the ACM* (1992), vol. 35(6): pp. 64–72 (cit. on pp. 44, 50).

CZERNUSZENKO, MAREK, DAVE PAPE, DANIEL SANDIN, TOM DEFANTI, GREGORY L DAWE and MAXINE D BROWN: 'The ImmersaDesk and Infinity Wall projection-based virtual reality displays'. *ACM SIGGRAPH Computer Graphics* (1997), vol. 31(2): pp. 46–49 (cit. on p. 51).

CZSENTMIHALYI, M: *Flow: The psychology of optimal experience*. 1990 (cit. on pp. 29, 44).

DA CUNHA, CATHERINE, BRUNO AGARD and ANDREW KUSIAK: 'Selection of modules for mass customisation'. *International Journal of Production Research* (2010), vol. 48(5): pp. 1439–1454 (cit. on pp. 82, 84).

DALY, JASON, BRYAN KLINE and GLENN A MARTIN: 'VESS: coordinating graphics, audio, and user interaction in virtual reality applications'. *Virtual Reality, 2002. Proceedings. IEEE*. IEEE. 2002: pp. 289–290 (cit. on p. 44).

DAMER, BRUCE, JEREMY JUDSON, JACKIE DOVE, STEVE ILLUSTRATOR-DIPAOLA, ALI ILLUSTRATOR-EBTEKAR, STASIA ILLUSTRATOR-MCGEHEE, RICHARD DESIGNED BY-WALKER and KATE PRODUCED BY-REBER: *Avatars!; Exploring and Building Virtual Worlds on the Internet*. Peachpit Press, 1997 (cit. on p. 43).

DANGELMAIER, WILHELM, MATTHIAS FISCHER, JÜRGEN GAUSEMEIER, MICHAEL GRAFE, CARSTEN MATYSCZOK and BENGT MUECK: 'Virtual and augmented reality support for discrete manufacturing system simulation'. *Computers in Industry* (2005), vol. 56(4): pp. 371–383 (cit. on pp. 40, 61).

DAS, SUMIT, ROGER ENGELMANN, RANDY HUDSON, TRINA ROY and LEWIS SIEGEL: 'Scientists in wonderland: A report on visualization applications in the CAVE virtual reality environment'. (1993), vol. (cit. on p. 52).

DAVIES, S and C MEYER: *Blur: the speed of change in the connected economy, Capstone*. 1998 (cit. on p. 82).

DAYAL, ABHINAV, CLIFF WOOLLEY, BENJAMIN WATSON and DAVID LUEBKE: 'Adaptive Frameless Rendering'. *ACM SIGGRAPH 2005 Courses*. SIGGRAPH '05. Los Angeles, California, 2005 (cit. on p. 50).

DEPARTMENT FOR BUSINESS, ENERGY & INDUSTRIAL STRATEGY (DBEIS): *Small Modular Reactors competition: phase one*. [Accessed 02 November 2017]. 2016. URL: https://www.gov.uk/government/publications/small-modular-reactors-competition-phase-one (cit. on p. 12).

DCNS: *Nuclear activities*. [Accessed 06 October 2014]. Direction des Constructions Navales Services. 2014. URL: http://en.dcnsgroup.com/energy/civil-nuclear-engineering/nuclear-activities/ (cit. on p. 14).

DECC: *Multimillion boost to UK economy as China and UK government sign civil nuclear agreement and sign agreement to deepen cooperation on climate change*. [Press Release]. [Accessed 03 September 2014]. 2014. URL: https://www.gov.uk/government/news/ (cit. on p. 9).

– *National Policy Statement for Nuclear Power Generation (EN-6)*. FactSheet. [Accessed 04 September 2014]. London, Stationary Office: Department of Energy & Climate Change, 2011. URL: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/47859/2009-nps-for-nuclear-volumeI.pdf (cit. on p. 10).

– *Planning our electric future: a White Paper for secure, affordable and low carbon electricity*. [White Paper]. [Accessed 03 September 2014]. Cm.8099. London, Stationary Office: Department of Energy & Climate Change, 2011. URL: https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/48129/2176-emr-white-paper.pdf (cit. on p. 9).

– *UK civil nuclear site map*. [Press Release]. [Accessed 04 September 2014]. 2014. URL: http://gov.uk (cit. on p. 11).

DEERING, MICHAEL: 'High resolution virtual reality'. *ACM SIGGRAPH Computer Graphics* (1992), vol. 26(2): pp. 195–202 (cit. on p. 50).

DELANO, WARREN L: 'The PyMOL molecular graphics system'. (2002), vol. (cit. on p. 52).

DELMIA: *Production System Simulation*. [Accessed 02 October 2014]. 2014. URL: http://www.3ds.com/products-services/delmia/products/manufacturing-planners/production-system-simulation/ (cit. on p. 61).

DELONE, WILLIAM H and EPHRAIM R MCLEAN: 'The DeLone and McLean model of information systems success: a ten-year update'. *Journal of management information systems* (2003), vol. 19(4): pp. 9–30 (cit. on p. 94).

DICK, ANTHONY R, PHILIP HS TORR and ROBERTO CIPOLLA: 'Modelling and interpretation of architecture from several images'. *International Journal of Computer Vision* (2004), vol. 60(2): pp. 111–134 (cit. on p. 60).

DIFEDE, JOANN, JUDITH CUKOR, IVY PATT, CEZAR GIOSAN and HUNTER HOFFMAN: 'The application of virtual reality to the treatment of PTSD following the WTC attack'. *Annals of the New York Academy of Sciences* (2006), vol. 1071(1): pp. 500–501 (cit. on p. 51).

DISSONANCE: *Dissonance VoIP*. [Accessed 17 April 2018]. 2018. URL: https://dissonance.readthedocs.io/en/latest/ (cit. on p. 145).

DEPARTMENT OF ENERGY (DOE): *Office of Nuclear Energy - Advanced Small Modular Reactors*. [Accessed 02 November 2017]. 2017. URL: https://www.energy.gov/ne/nuclear-reactor-technologies/small-modular-nuclear-reactors (cit. on p. 13).

DOMAGOJ BARIČEVIĆ Hrvoje Dujmić, MATKO ŠARIĆ: *QAVE: a Game-based Immersive Virtual Reality System*. [Accessed 01 October 2014]. 2007. URL: http://marjan.fesb.hr/qave/pubs/QAVE%5C%20-%5C%20IIS%5C%202007.pdf (cit. on p. 53).

DOMAH, PREETUM and MICHAEL G POLLITT: *The restructuring and privatisation of electricity distribution and supply businesses in England and Wales: a social cost benefit analysis*. Department of Applied Economics, University of Cambridge, 2000 (cit. on p. 9).

DOUGLAS, YELLOWLEES and ANDREW HARGADON: 'The pleasure principle: immersion, engagement, flow'. *Proceedings of the eleventh ACM on Hypertext and hypermedia*. ACM. 2000: pp. 153–160 (cit. on pp. 29, 44).

DUCHOWSKI, ANDREW T, VINAY SHIVASHANKARAIAH, TIM RAWLS, ANAND K GRAMOPADHYE, BRIAN J MELLOY and BARBARA KANKI: 'Binocular eye tracking in virtual reality for inspection training'. *Proceedings of the 2000 symposium on Eye tracking research & applications*. ACM. 2000: pp. 89–96 (cit. on p. 45).

EBERHARD, P and Z LI: 'Virtual reality simulation of multibody systems'. *Proceedings of EUROMECH colloquium*. Vol. 476. 2006 (cit. on p. 59).

ENERGY DE FRANCE (EDF): *Blog: Milestones Achieved at Hinkley Point C – First Nuclear Concrete Pour*. [Accessed 02 November 2017]. 2017. URL: https://www.edfenergy.com/energy/nuclear-new-build-projects/hinkley-point-c/news-views/concrete-pour-march-2017 (cit. on p. 8).

EDF-ENERGY: *EDF Energy's nuclear power stations*. [Online]. [Accessed 04 September 2014]. 2014. URL: http://www.edfenergy.com/energyfuture/edf-energys-approach-why-we-choose-new-nuclear/current-nuclear-sites (cit. on p. 8).

EDF-ENERGY: *Nuclear New Build projects*. [Online]. [Accessed 03 Sep. 2014]. 2014. URL: http://www.edfenergy.com/energy/nuclear-new-build-projects (cit. on p. 11).

EISLER, RD, AK CHATTERJEE, D VASKE and GH BURGHART: *Phase 1. Development of an Urban Warfare Virtual Environment. Modeling and Simulation of Weapons Effects on Building Structural Integrity and Personnel for DIS Virtual Environments.* Tech. rep. DTIC Document, 1996 (cit. on p. 56).

EISLER, RD, G TYRA, AK CHATTERJEE and GH BURGHART: *Development of an Urban Warfare Virtual Environment*. Tech. rep. DTIC Document, 2001 (cit. on p. 43).

EL HAOUZI, HIND, ANDRÉ THOMAS and JEAN-FRANÇOIS PÉTIN: 'Contribution to reusability and modularity of manufacturing systems simulation models: Application to distributed control simulation within DFT context'. *International Journal of Production Economics* (2008), vol. 112(1): pp. 48–61 (cit. on p. 84).

ELMARAGHY, H, T ALGEDDAWY, A AZAB and W ELMARAGHY: 'Change in Manufacturing-Research and Industrial Challenges'. *Enabling Manufacturing Competitiveness and Economic Sustainability*. Springer, 2012: pp. 2–9 (cit. on pp. 56, 60).

ELMARAGHY, HODA A and TAREK ALGEDDAWY: 'Co-evolution of products and manufacturing capabilities and application in auto-parts assembly'. *Flexible services and manufacturing journal* (2012), vol. 24(2): pp. 142–170 (cit. on p. 83).

EMMELKAMP, PAUL MG, MARY BRUYNZEEL, LEONIE DROST and CHARLES AP G van der MAST: 'Virtual reality treatment in acrophobia: a comparison with exposure in vivo'. *CyberPsychology & Behavior* (2001), vol. 4(3): pp. 335–339 (cit. on p. 51).

ENCORD: *29th meeting ENCORD Virtual Construction Platform April 2014*. [Accessed 02 November 2015]. 2014. URL: http://www.encord.org/?cat=8 (cit. on p. 75).

ENGINEERING-VILLAGE: *Engineering Village Compendex & Geobase databases*. [Online]. [Accessed 29 September 2014]. 2012. URL: http://www.engineeringvillage2.org (cit. on p. 64).

EPIC: *Unreal Engine 4*. [Accessed 17 April 2018]. 2018. URL: https://www.unrealengine.com/en-US/what-is-unreal-engine-4 (cit. on p. 171).

F-M: *france-Metallurgie*. [Online]. [Accessed 03 Sep. 2014]. 2014. URL: http://www.france-metallurgie.com (cit. on p. 16).

FABRA, UNIVERSITAT POMPEU: *Impart - Intelligent Management Platform for Advanced Real-Time media processes*. [Accessed 17 April 2018]. 2018. URL: https://impart.upf.edu/ (cit. on p. 139).

FAN, JEFF C and KWOTING FANG: 'ERP implementation and information systems success: a test of DeLone and McLean's model'. *Technology Management for the Global Future, 2006. PICMET 2006*. Vol. 3. IEEE. 2006: pp. 1272–1278 (cit. on p. 94).

FDA: *Computer-Assisted Surgical Systems*. [Accessed 12 February 2019]. 2018. URL: https://www.fda.gov/medicaldevices/productsandmedicalprocedures/surgeryandlifesupport/computerassistedsurgicalsystems/default.htm (cit. on p. 51).

– *FDA clears new robotically-assisted surgical device for adult patients*. [Accessed 12 February 2019]. 2017. URL: https://www.fda.gov/newsevents/newsroom/pressannouncements/ucm580452.htm (cit. on p. 51).
FECHTELER, PHILIPP, PETER EISERT and JÜRGEN RURAINSKY: 'Fast and high resolution 3d face scanning'. *Image Processing, 2007. ICIP 2007. IEEE International Conference on*. Vol. 3. IEEE. 2007: pp. III–81 (cit. on p. 60).
FESTO: *What functions can Industry 4.0 access here and now?* [Online]. [Accessed 23 October 2017]. 2013. URL: http://cdn.iotwfhttps://www.festo.com/net/it-ch%5C_ch/SupportPortal/Details/280102/PressArticle.aspx (cit. on p. 92).
FETTKE, PETER and PETER LOOS: 'Referenzmodellierungsforschung'. *Wirtschaftsinformatik* (2004), vol. 46(5): pp. 331–340 (cit. on p. 87).
FISHER, SCOTT: 'Viewpoint dependent imaging: An interactive stereoscopic display'. *26th Annual Technical Symposium*. International Society for Optics and Photonics. 1983: pp. 41–45 (cit. on p. 49).
FISHER, SCOTT, MICHEAL MCGREEVY, JAMES HUMPHRIES and WARREN ROBINETT: 'Virtual environment display system'. *Proceedings of the 1986 workshop on Interactive 3D graphics*. ACM. 1987: pp. 77–87 (cit. on p. 49).
FLOCK, HOWARD R: 'Optical texture and linear perspective as stimuli for slant perception.' *Psychological review* (1965), vol. 72(6): p. 505 (cit. on p. 54).
FOGLIATTO, FLAVIO S, GIOVANI JC DA SILVEIRA and DENIS BORENSTEIN: 'The mass customization decade: An updated review of the literature'. *International Journal of Production Economics* (2012), vol. 138(1): pp. 14–25 (cit. on pp. 82, 84, 94).
FRANKE, R: 'Kopplung von diskreter Simulation und interaktiver 3D-Visualisierung'. MSc. Germany: University of Magdeburg, 2004 (cit. on p. 63).
FRAUNHOFER, INST: *Digital Production*. [Online]. [Accessed 27 September 2014]. 2014. URL: http://www.fraunhofer.de/en/institutes-research-establishments/innovation-clusters/digital-production.html (cit. on p. 59).
FRAZZON, ENZO MOROSINI, JENS HARTMANN, THOMAS MAKUSCHEWITZ and BERND SCHOLZ-REITER: 'Towards socio-cyber-physical systems in production networks'. *Procedia Cirp* (2013), vol. 7: pp. 49–54 (cit. on pp. 84, 97).
FREEMAN, C., R. SCOTT and S. REDDISH: 'Discrete Event Simulation in Immersive Virtual Reality'. *Proceedings of the 4th Joint Virtual Reality Conference ICAT-EGVE-EuroVR*. JVRC '12. Madrid, Spain: Eurographics Association, 2012 (cit. on p. 67).
FRIEDEN, ROY: *Probability, statistical optics, and data testing: a problem solving approach*. Vol. 10. Springer, 2001 (cit. on p. 230).
FINANCIAL TIMES (FT): *Hedge funds place $5.4bn bet on Toshiba's resurrection*. [Accessed 02 November 2017]. 2017. URL: https://www.ft.com/content/6cb938ca-cdc5-11e7-b781-794ce08b24dc (cit. on p. 13).
FUCHS, KH: 'Minimally invasive surgery'. *Endoscopy* (2002), vol. 34(02): pp. 154–159 (cit. on p. 51).
FUGAZZOLA, LAURA, SILVANA PILOTTI, ALDO PINCHERA, TATIANA V VORONTSOVA, PIERA MONDELLINI, ANGELA GRECO, LARISA ASTAKHOVA, MARTA G BUTTI, EUGENE P DEMIDCHIK, FURIO PACINI et al.: 'Oncogenic rearrangements of the RET

proto-oncogene in papillary thyroid carcinomas from children exposed to the Chernobyl nuclear accident'. *Cancer Research* (1995), vol. 55(23): pp. 5617–5620 (cit. on p. 18).

FURNESS, T: 'Designing in virtual space'. *System design: behavioral perspectives on designers, tools, and organizations. North Holland, New York* (1987), vol.: pp. 127–143 (cit. on p. 49).

GAITATZES, ATHANASIOS, DIMITRIOS CHRISTOPOULOS and MARIA ROUSSOU: 'Reviving the Past: Cultural Heritage Meets Virtual Reality'. *Proceedings of the 2001 Conference on Virtual Reality, Archeology, and Cultural Heritage*. VAST '01. Glyfada, Greece: ACM, 2001: pp. 103–110 (cit. on p. 52).

GE: *Digital Twin & Machine Learning - The Differentiator in Oil and Gas IIoT*. [Online]. [Accessed 23 October 2017]. 2018. URL: `https://www.geoilandgas.com/software-solutions/industrial-internet/digital-twin-machine-learning-differentiator-oil-and-gas-iiot` (cit. on p. 91).

GEISBERGER, EVA and MANFRED BROY: *agendaCPS: Integrierte Forschungsagenda Cyber-Physical Systems*. Vol. 1. Springer-Verlag, 2012 (cit. on p. 87).

GEISELHART, FLORIAN, MICHAEL OTTO and ENRICO RUKZIO: 'On the use of multi-depth-camera based motion tracking systems in production planning environments'. *Procedia CIRP* (2016), vol. 41: pp. 759–764 (cit. on pp. 134, 169).

GERARDI, MARYROSE, BARBARA OLASOV ROTHBAUM, KERRY RESSLER, MARY HEEKIN and ALBERT RIZZO: 'Virtual reality exposure therapy using a virtual Iraq: case report'. *Journal of traumatic stress* (2008), vol. 21(2): pp. 209–213 (cit. on p. 51).

GIBSON, I, D BROWN, S COBB and R EASTGATE: 'VIRTUAL REALITY AND RAPID PROTOTYPING: CONFLICTING OR COMPLIMENTARY!' *Proc. 4th Solid Freeform Fabrication Symp*. 1993: pp. 113–121 (cit. on p. 56).

GIGABYTE: *Mni-PcBarebone*. [Accessed 17 April 2018]. 2018. URL: `https://www.gigabyte.com/Mini-PcBarebone` (cit. on p. 120).

GISI, MARK A and CRISTIANO SACCHI: 'Co-CAD- A collaborative mechanical CAD system'. *Presence- Teleoperators and Virtual Environments* (1994), vol. 3(4): pp. 341–350 (cit. on p. 59).

GLOBERSON, SHLOMO, NISSAN LEVIN and AVRAHAM SHTUB: 'The Impact of Breaks on Forgetting When Performing A Repetitive Task'. *IIE Transactions* (1989), vol. 21(4): pp. 376–381 (cit. on pp. 15, 16).

GENERAL NUCLEAR SYSTEM LIMITED (GNS): *Regulators start technical assessment of UK HPR 1000*. [Accessed 02 November 2017]. 2017. URL: `http://www.ukhpr1000.co.uk/` (cit. on p. 10).

GOMES DE SÁ, ANTONINO and GABRIEL ZACHMANN: 'Virtual reality as a tool for verification of assembly and maintenance processes'. *Computers & Graphics* (1999), vol. 23(3): pp. 389–403 (cit. on p. 66).

GÓMEZ-PÉREZ, ASUNCIÓN, MARIANO FERNÁNDEZ-LÓPEZ and OSCAR CORCHO: *Ontological Engineering: with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*. Springer Science & Business Media, 2006 (cit. on pp. 88, 89).

GOODFELLOW, MARTIN J, HUGO R WILLIAMS and ADISA AZAPAGIC: 'Nuclear renaissance, public perception and design criteria: An exploratory review'. *Energy Policy* (2011), vol. 39(10): pp. 6199–6210 (cit. on p. 9).

GOOGLE: *Google TiltBrush*. [Accessed 17 April 2018]. 2018. URL: `https://www.tiltbrush.com/` (cit. on pp. 59, 175).

– *Material Design System*. [Accessed 17 April 2018]. 2018. URL: https://material.io/design/ (cit. on p. 150).

UK-GOV: *Economic Benefit of Improving the UK's Nuclear Supply Chain Capabilities*. [Online]. [Accessed 26 March 2013]. 2013. URL: https://www.gov.uk/government/publications/economic-benefit-of-improving-the-uks-nuclear-supply-chain-capabilities (cit. on p. 20).

– *Economic Benefit of Improving the UK's Nuclear Supply Chain Capabilities*. [Online]. [Accessed 26 March 2013]. 2013. URL: https://www.gov.uk/government/publications/nuclear-industrial-vision-statement (cit. on p. 20).

– *Nuclear Industrial Strategy*. [Online]. [Accessed 26 March 2013]. 2013. URL: https://www.gov.uk/government/organisations/department-for-business-innovation-skills/series/nuclear-industrial-strategy (cit. on p. 20).

– *Nuclear Industrial Vision Statement*. [Online]. [Accessed 26 March 2013]. 2013. URL: https://www.gov.uk/government/publications/nuclear-industrial-vision-statement (cit. on p. 20).

GOVERNMENT, UK: *Sizewell B Inquiry, 1920-1987*. [Online]. Public Enquiry. [Accessed 08 August 2014]. 2014. URL: http://discovery.nationalarchives.gov.uk/details/r/C7081%E2%80%8E (cit. on p. 8).

GRIEVES, MICHAEL: 'Digital Twin: Manufacturing excellence through virtual factory replication'. *White paper* (2014), vol. (cit. on pp. 90, 112).

– *Virtually perfect: Driving innovative and lean products through product lifecycle management*. Space Coast Press, 2011 (cit. on p. 90).

GRIMES, ROBIN W and WILLIAM J NUTTALL: 'Generating the option of a two-stage nuclear renaissance'. *Science* (2010), vol. 329(5993): pp. 799–803 (cit. on p. 9).

GRUBER, TR: 'A translation approach to portable ontologies Knowledge Acquisition 5 (2) 199220'. (1993), vol. (cit. on p. 89).

GRUENING, JAMES, JAMES BERNARD, CHRIS CLOVER and KURT HOFFMEISTER: *Driving simulation*. Tech. rep. SAE Technical Paper, 1998 (cit. on p. 51).

GUENDELMAN, ERAN, ROBERT BRIDSON and RONALD FEDKIW: 'Nonconvex Rigid Bodies with Stacking'. *ACM Trans. Graph.* (July 2003), vol. 22(3): pp. 871–878 (cit. on p. 57).

GUPTA, MOHIT, AMIT AGRAWAL, ASHOK VEERARAGHAVAN and SRINIVASA G NARASIMHAN: 'Structured light 3D scanning in the presence of global illumination'. *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*. IEEE. 2011: pp. 713–720 (cit. on p. 60).

HAASE, HELMUT: 'Symbiosis of virtual reality and scientific visualization system'. *Computer Graphics Forum*. Vol. 15. Wiley Online Library. 1996: pp. 443–451 (cit. on p. 52).

HAGGAG, HUSSEIN, MOHAMMED HOSSNY, SAEID NAHAVANDI and D CREIGHTON: 'Real Time Ergonomic Assessment for Assembly Operations Using Kinect'. *Computer Modelling and Simulation (UKSim), 2013 UKSim 15th International Conference on*. IEEE. 2013: pp. 495–500 (cit. on p. 66).

HALLER, STEPHAN, STAMATIS KARNOUSKOS and CHRISTOPH SCHROTH: 'The internet of things in an enterprise context'. *Future Internet Symposium*. Springer. 2008: pp. 14–28 (cit. on p. 94).

HALLNÄS, LARS and JOHAN REDSTRÖM: 'Slow technology-designing for reflection'. *Personal and ubiquitous computing* (2001), vol. 5(3): pp. 201–212 (cit. on p. 24).

HAMDI, MUSTAPHA, ANTOINE FERREIRA, GAURAV SHARMA and CONSTANTINOS MAVROIDIS: 'Prototyping bio-nanorobots using molecular dynamics simulation and virtual reality'. *Microelectronics Journal* (2008), vol. 39(2): pp. 190–201 (cit. on p. 52).

HEALEY, GLEEN and THOMAS O BINFORD: 'Local shape from specularity'. *Computer Vision, Graphics, and Image Processing* (1988), vol. 42(1): pp. 62–86 (cit. on p. 60).

HEILIG, MORTON L: 'El Cine del Futuro: the cinema of the future'. *Presence: Teleoperators and Virtual Environments* (1992), vol. 1(3): pp. 279–294 (cit. on pp. 44, 45).

– *Sensorama simulator*. US Patent 3,050,870. Nov. 1962 (cit. on pp. 46, 47).

– *Stereoscopic-television apparatus for individual use*. US Patent 2,955,156. Oct. 1960 (cit. on pp. 46, 47).

HENNING, KAGERMANN: 'Recommendations for implementing the strategic initiative INDUSTRIE 4.0'. (2013), vol. (cit. on p. 79).

HENRIKSEN, JIM: *Proof Animation*. [Accessed 30 September 2014]. Wolverine Software. 2014. URL: http://www.wolverinesoftware.com/ (cit. on pp. 61, 63).

– *SLX simulation*. [Accessed 30 September 2014]. Wolverine Software. 2014. URL: http://www.wolverinesoftware.com/ (cit. on p. 63).

HIBINO, H, T INUKAI and Y FUKUDA: 'Efficient manufacturing system implementation based on combination between real and virtual factory'. *International Journal of Production Research* (2006), vol. 44(18-19): pp. 3897–3915 (cit. on pp. 61, 63).

HITACHI-GE: *UK Advanced Boiling Water Reactor*. [Online]. [Accessed 04 September 2014]. 2014. URL: http://www.hitachi-hgne-uk-abwr.co.uk/ (cit. on p. 10).

HITCHNER, LEWIS E: 'Virtual planetary exploration: A very large virtual environment'. *ACM SIGGRAPH*. Vol. 92. 1992 (cit. on p. 52).

HITOMI, KATSUNDO: *Manufacturing Systems Engineering: A Unified Approach to Manufacturing Technology, Production Management and Industrial Economics*. Routledge, 2017 (cit. on p. 89).

HODGES, LARRY F, ROB KOOPER, THOMAS C MEYER, BARBARA O ROTHBAUM, DAN OPDYKE, JOHANNES J de GRAAFF, JAMES S WILLIFORD and MAX M NORTH: 'Virtual environments for treating the fear of heights'. *Computer* (1995), vol. 28(7): pp. 27–34 (cit. on p. 51).

HOLLOWAY, RICHARD and ANSELMO LASTRA: 'Virtual environments: A survey of the technology'. *SIGGRAPH 95 course*. A 8. Eurographics. 1995: pp. 1–40 (cit. on p. 44).

HORIZON: *Horizon Nuclear Power*. [Online]. [Accessed 03 Sep. 2014]. 2014. URL: http://www.horizonnuclearpower.com/ (cit. on p. 11).

HORN, BERTHOLD KP and MICHAEL J BROOKS: *Shape from shading*. MIT press, 1989 (cit. on p. 60).

HOWARD, IAN P and BRIAN J ROGERS: *Binocular vision and stereopsis*. Oxford University Press, USA, 1995 (cit. on p. 54).

HOWLETT, ERIC M: 'Wide-angle orthostereo'. *SC-DL tentative*. International Society for Optics and Photonics. 1990: pp. 210–223 (cit. on p. 49).

HSE: *General Design Assessments*. [Online]. [Accessed 30 March 2013] Avaliable from: 2013. URL: http://www.hse.gov.uk/newreactors/ (cit. on p. 10).

HUXLEY, ALDOUS: *Brave New World*. Ernst Klett Sprachen, 1932 (cit. on p. 45).

IAEA: *Appendix VI - IAEA International Nuclear Event Scale (INES)*. Appendices. [Accessed 04 September 2014]. Vienna: International Atomic Energy Authority, 2011. URL: http://www.iaea.org/ns/tutorials/regcontrol/appendix/app96.htm (cit. on p. 9).

– *Evaluated Nuclear Data File (ENDF)*. [Online]. [Accessed 03 September 2014]. 2009. URL: https://www-nds.iaea.org/exfor/endf.htm (cit. on p. 19).

– *Lessons Learned the Hard Way*. [Accessed 06 October 2014]. International Atomic Energy Association. 2014. URL: http://www.iaea.org/Publications/Magazines/Bulletin/Bull472/htmls/lessons_learned.html (cit. on p. 19).

IFIXIT: *Xbox One Kinect Teardown*. [Accessed 17 April 2018]. 2018. URL: https://www.ifixit.com/Teardown/Xbox+One+Kinect+Teardown/19725 (cit. on p. 119).

INPO: *Institute of Nuclear Power Operations*. [Accessed 08 October 2014]. 2014. URL: http://www.inpo.info/ (cit. on p. 18).

ISDA: *Polyglot*. [Accessed 08 October 2014]. University of Illinois Urbana-Chapaign, Image and Spatial Data Analysis Division. 2014. URL: http://isda.ncsa.illinois.edu/drupal/software/polyglot (cit. on p. 71).

JACKS, EDWIN L: 'A laboratory for the study of graphical man-machine communication'. *Proceedings of the October 27-29, 1964, fall joint computer conference, part I*. ACM. 1964: pp. 343–350 (cit. on pp. 46, 56).

JACOBSON, JEFFREY: *CaveUT*. [Accessed 01 October 2014]. Wolverine Software. 2005. URL: http://publicvr.org/html/pro_caveut.html (cit. on p. 52).

JACOBSON, JEFFREY and ROBYN GILLAM: 'The Egyptian Oracle; Recreating Egyptian Religious Ceremony in Mixed Reality'. (2012), vol. [Accessed 01 October 2014]. URL: http://publicvr.org/publications/JOCCH-7.pdf (cit. on p. 53).

JACOBSON, JEFFREY and MICHAEL LEWIS: 'Game engine virtual reality with CaveUT'. *Computer* (2005), vol. 38(4): pp. 79–82 (cit. on p. 39).

JAEHNE, DM, MIAO LI, RALPH RIEDEL and EGON MUELLER: 'Configuring and operating global production networks'. *International Journal of Production Research* (2009), vol. 47(8): pp. 2013–2030 (cit. on p. 82).

JAGUARLANDROVER: *VNR Evoque Technical Seminars - Virtual Reality 3D Cave Vodcast*. [Accessed 02 November 2015]. 2011. URL: http://newsroom.jaguarlandrover.com/en-in/land-rover/videos//video?id=10484 (cit. on p. 75).

JAYARAM, S., U. JAYARAM, Y. WANG, H. TIRUMALI, K. LYONS and P. HART: 'VADE: A Virtual Assembly Design Environment'. *IEEE Computer Graphics and Applications* (1999), vol. 19(6): pp. 44–50 (cit. on p. 64).

JAYARAM, S., UMA JAYARAM, YOUNG JUN KIM, CHARLES DECHENNE, KEVIN W LYONS, CRAIG PALMER and TATSUKI MITSUI: 'Industry case studies in the use of immersive virtual assembly'. *Virtual Reality* (2007), vol. 11(4): pp. 217–228 (cit. on pp. 65, 68).

JAZDI, NASSER: 'Cyber physical systems in the context of Industry 4.0'. *Automation, Quality and Testing, Robotics, 2014 IEEE International Conference on*. IEEE. 2014: pp. 1–4 (cit. on p. 87).

JING, QI, ATHANASIOS V VASILAKOS, JIAFU WAN, JINGWEI LU and DECHAO QIU: 'Security of the internet of things: Perspectives and challenges'. *Wireless Networks* (2014), vol. 20(8): pp. 2481–2501 (cit. on p. 92).

JOSH BLAKE, KYLE MACHULIS: *libfreenect GitHub*. [Accessed 17 April 2018]. 2018. URL: https://github.com/OpenKinect/libfreenect (cit. on p. 119).

JUAREZ, ALEX, WILLEM SCHONENBERG and CHRISTOPH BARTNECK: 'Implementing a low-cost CAVE system using the CryEngine2'. *Entertainment Computing* (2010), vol. 1(3): pp. 157–164 (cit. on p. 53).

JUN, YAO, JIANHUA LIU, RUXIN NING and YE ZHANG: 'Assembly process modeling for virtual assembly process planning'. *International Journal of Computer Integrated Manufacturing* (2005), vol. 18(6): pp. 442–451 (cit. on p. 64).

KAGERMANN, HENNING, JOHANNES HELBIG, ARIANE HELLINGER and WOLFGANG WAHLSTER: *Recommendations for implementing the strategic initiative INDUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group*. Forschungsunion, 2013 (cit. on p. 80).

– *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0: Deutschlands Zukunft als Produktionsstandort sichern; Abschlussbericht des Arbeitskreises Industrie 4.0*. Forschungsunion, 2013 (cit. on p. 94).

KALAWSKY, ROY: *The Science of Virtual Reality and Virtual Environments*. 1st. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1993 (cit. on p. 57).

KAPLAN, ELLIOTT D and CHRISTOPHER J HEGARTY: *Understanding GPS: principles and applications*. Artech house, 2005 (cit. on p. 228).

KARKEE, MANOJ, BRIAN L STEWARD, ATUL G KELKAR and ZACHARY T KEMP II: 'Modeling and real-time simulation architectures for virtual prototyping of off-road vehicles'. *Virtual reality* (2011), vol. 15(1): pp. 83–96 (cit. on p. 59).

KAUFMANN, HANNES, DIETER SCHMALSTIEG and MICHAEL WAGNER: 'Construct3D: a virtual reality application for mathematics and geometry education'. *Education and information technologies* (2000), vol. 5(4): pp. 263–276 (cit. on p. 65).

KAWASAKI, HARUHISA, TETSUYA MOURI, M OSAMA ALHALABI, YASUTAKA SUGIHASHI, YOSHIO OHTUKA, SHO IKENOHATA, KAZUSHIGE KIGAKU, VYTAUTAS DANIULAITIS, KAZUYASU HAMADA and TATSUO SUZUKI: 'Development of five-fingered haptic interface: HIRO-II'. *Proceedings of the 2005 international conference on Augmented tele-existence*. ACM. 2005: pp. 209–214 (cit. on p. 233).

KELSICK, JASON J and JUDY M VANCE: 'The VR factory: discrete event simulation implemented in a virtual environment'. *Proceedings of DETC'98*. ASME Design Engineering Technical Conferences. 1998 (cit. on pp. 40, 61).

KELSICK, JASON, JUDY M VANCE, LORI BUHR and CHERYL MOLLER: 'Discrete event simulation implemented in a virtual environment'. *Journal of Mechanical Design* (2003), vol. 125(3): pp. 428–433 (cit. on p. 61).

KENDER, JOHN R: 'Shape from texture: An aggregation transform that maps a class of textures into surface orientation'. *Proceedings of the 6th international joint conference on Artificial intelligence-Volume 1*. Morgan Kaufmann Publishers Inc. 1979: pp. 475–480 (cit. on p. 60).

KHALAF, RADWAN EL HADJ, BRUNO AGARD and BERNARD PENZ: 'An experimental study for the selection of modules and facilities in a mass customization context'. *Journal of Intelligent Manufacturing* (2010), vol. 21(6): pp. 703–716 (cit. on pp. 82, 84).

– ‘Simultaneous design of a product family and its related supply chain using a Tabu Search algorithm’. *International Journal of Production Research* (2011), vol. 49(19): pp. 5637–5656 (cit. on p. 83).

KIBIRA, DEOGRATIAS and CHUCK MCLEAN: ‘Manufacturing Modeling Methods: Virtual Reality Simulation of a Mechanical Assembly Production Line’. *Proceedings of the 34th Conference on Winter Simulation: Exploring New Frontiers*. WSC ’02. San Diego, California: Winter Simulation Conference, 2002: pp. 1130–1137 (cit. on p. 62).

KIM, VICTOR B, WILLIAM HH CHAPMAN III, ROBERT J ALBRECHT, B MARCUS BAILEY, JAMES A YOUNG, L WILEY NIFONG and W RANDOLPH CHITWOOD JR: ‘Early experience with telemanipulative robot-assisted laparoscopic cholecystectomy using da Vinci’. *Surgical Laparoscopy Endoscopy & Percutaneous Techniques* (2002), vol. 12(1): pp. 33–40 (cit. on p. 51).

KIM, W.S.: ‘Virtual reality calibration for telerobotic servicing’. *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*. May 1994: 2769–2775 vol.4 (cit. on p. 72).

KLÖPPER, BENJAMIN, JAN PATRICK PATER and WILHELM DANGELMAIER: ‘Parallel scheduling for evolving manufacturing systems’. *Industrial Informatics (INDIN), 2012 10th IEEE International Conference on*. IEEE. 2012: pp. 1086–1091 (cit. on p. 87).

KNUTZON, JARED S, BRYAN E WALTER, ADRIAN V SANNIER and JAMES H OLIVER: ‘An immersive approach to command and control’. *Journal of Battlefield Technology* (2004), vol. 7(1): p. 37 (cit. on p. 55).

KOCH, MARGARETE T, HENNING BAARS, HEINER LASI and HANS-GEORG KEMPER: ‘Manufacturing Execution Systems and Business Intelligence for Production Environments.’ *AMCIS*. 2010: p. 436 (cit. on p. 87).

KOLASINSKI, EUGENIA M: *Simulator Sickness in Virtual Environments.* Tech. rep. DTIC Document, 1995 (cit. on p. 52).

KREYLOS, OLIVER: *Good Stereo Vs Bad Stereo*. [Online]. [Accessed 11 Sep. 2014]. 2014. URL: http://doc-ok.org/?p=77 (cit. on p. 28).

KRIJN, MEREL, PAUL M.G EMMELKAMP, ROELINE BIEMOND, CLAUDIUS de WILDE DE LIGNY, MARTIJN J SCHUEMIE and CHARLES A.P.G van der MAST: ‘Treatment of acrophobia in virtual reality: The role of immersion and presence’. *Behaviour Research and Therapy* (2004), vol. 42(2): pp. 229–239 (cit. on p. 44).

KROGH, BRUCE H: ‘Cyber Physical Systems: the need for new models and design paradigms’. *Presentation Report* (2008), vol. (cit. on p. 84).

KU, JEONGHUN, RICHARD MRAZ, NICOLE BAKER, KONSTANTINE K ZAKZANIS, JANG HAN LEE, IN Y KIM, SUN I KIM and SIMON J GRAHAM: ‘A data glove with tactile feedback for FMRI of virtual reality experiments’. *Cyberpsychology & behavior* (2003), vol. 6(5): pp. 497–508 (cit. on p. 52).

KUNTZ, SEBASTIEN: *MiddleVR*. [Accessed 01 October 2014]. Im in VR. 2014. URL: http://www.imin-vr.com/middlevr/ (cit. on p. 52).

L3: *L3-MAPPS POWER SYSTEMS AND SIMULATION*. [Accessed 01 December 2018]. 2015. URL: https://www2.l3t.com/mapps/en/products-and-services-power.html# (cit. on p. 4).

LANIER, JARON: *Interview with Jaron Lanier & Kevin Kelly*. [Interview]. [Accessed 24 September 2014]. 2014. URL: http://www.theverge.com/a/virtual-reality/interview (cit. on p. 55).

– *VPL software Lives*. [Online]. [Accessed 25 September 2014]. 2010. URL: http://www.jaronlanier.com/vpl.html (cit. on pp. 49, 50).

LANNER: *Visionary Render™ For WITNESS™ Launched*. [Accessed 02 November 2017]. 2015. URL: https://www.lanner.com/insights/news/visionary-render-for-witness-launched.html (cit. on p. 61).

LAPERRIÈRE, L. and H.A. ELMARAGHY: 'Assembly sequences planning for simultaneous engineering applications'. *The International Journal of Advanced Manufacturing Technology* (1994), vol. 9(4): pp. 231–244 (cit. on p. 64).

LASI, H, D MORAR and HG KEMPER: 'Additive Manufacturing–Herausforderungen für die gestaltungsorientierte Wirtschaftsinformatik'. *Tagungsband Multikonferenz Wirtschaftsinformatik* (2014), vol.: pp. 417–418 (cit. on p. 87).

LASI, HEINER, PETER FETTKE, HANS-GEORG KEMPER, THOMAS FELD and MICHAEL HOFFMANN: 'Industry 4.0'. *Business & Information Systems Engineering* (2014), vol. 6(4): pp. 239–242 (cit. on pp. 82, 84).

LAU, HIU-FAI, KUNG-WONG LAU and CHI-WAI KAN: 'The Future of Virtual Environments: The Development of Virtual Technology'. *Computer Science and Information Technology* (2013), vol. 1: pp. 41–50 (cit. on p. 49).

LEE, EDWARD A: 'CPS foundations'. *Design Automation Conference (DAC), 2010 47th ACM/IEEE*. IEEE. 2010: pp. 737–742 (cit. on p. 85).

– 'Cyber physical systems: Design challenges'. *Object oriented real-time distributed computing (isorc), 2008 11th ieee international symposium on*. IEEE. 2008: pp. 363–369 (cit. on pp. 84–87, 182).

– 'The problem with threads'. *Computer* (2006), vol. 39(5): pp. 33–42 (cit. on p. 87).

LEE, J and E LAPIRA: 'Predictive factories: the next transformation'. *Manufacturing Leadership Journal* (2013), vol. 20(1): pp. 13–24 (cit. on p. 84).

LEE, JAY, BEHRAD BAGHERI and HUNG-AN KAO: 'A cyber-physical systems architecture for industry 4.0-based manufacturing systems'. *Manufacturing Letters* (2015), vol. 3: pp. 18–23 (cit. on pp. 84, 85).

LEE, JAY, EDZEL LAPIRA, BEHRAD BAGHERI and HUNG-AN KAO: 'Recent advances and trends in predictive manufacturing systems in big data environment'. *Manufacturing Letters* (2013), vol. 1(1): pp. 38–41 (cit. on p. 84).

LEE, JAY, EDZEL LAPIRA, SHANHU YANG and ANN KAO: 'Predictive Manufacturing System-Trends of Next-Generation Production Systems'. *IFAC Proceedings Volumes* (2013), vol. 46(7): pp. 150–156 (cit. on p. 84).

LEE, TAEKI and K HARRISON: 'Assessing safety culture in nuclear power stations'. *Safety science* (2000), vol. 34(1): pp. 61–97 (cit. on p. 19).

LEHNER, VALERIE D and THOMAS A DEFANTI: 'Distributed virtual reality: Supporting remote collaboration in vehicle design'. *Computer Graphics and Applications, IEEE* (1997), vol. 17(2): pp. 13–17 (cit. on p. 59).

LEICA-GEOSYSTEMS: *Laser Scanners*. [Accessed 17 April 2018]. 2018. URL: https://leica-geosystems.com/en-gb/products/laser-scanners (cit. on p. 138).

LEU, MING C, HODA A ELMARAGHY, ANDREW YC NEE, SOH KHIM ONG, MICHELE LANZETTA, MATTHIAS PUTZ, WENJUAN ZHU and ALAIN BERNARD: 'CAD model based virtual assembly simulation, planning and training'. *CIRP Annals-Manufacturing Technology* (2013), vol. 62(2): pp. 799–822 (cit. on pp. 58, 75).

LEVY, RICHARD M: 'The Virtual Reality Revolution: The Vision and the Reality'. (2012), vol. (cit. on p. 52).

LI, JING RONG, LI PHENG KHOO and SHU BENG TOR: 'An object-oriented intelligent disassembly sequence planner for maintenance'. *Computers in Industry* (2005), vol. 56(7): pp. 699–718 (cit. on p. 66).

LIAN, Y-H and HENDRIK VAN LANDEGHEM: 'Analysing the effects of Lean manufacturing using a value stream mapping-based simulation generator'. *International Journal of Production Research* (2007), vol. 45(13): pp. 3037–3058 (cit. on p. 84).

LIAO, H., L. ZHANG, T. XIAO, L. ZENG and Y. GU: 'Interactive assembly process planner'. *Tsinghua Science and Technology* (2004), vol. 9(2): pp. 219–226 (cit. on p. 65).

LIN, FUHUA, LAN YE, VINCENT G DUFFY and CHUAN-JUN SU: 'Developing virtual environments for industrial training'. *Information Sciences* (2002), vol. 140(1–2). Interactive virtual environments and distance education: pp. 153–170. URL: http://www.sciencedirect.com/science/article/pii/S0020025501001852 (cit. on p. 65).

LIN, HAO W, SEV V NAGALINGAM, SWEE S KUIK and TOMOHIRO MURATA: 'Design of a global decision support system for a manufacturing sme: towards participating in collaborative manufacturing'. *International Journal of Production Economics* (2012), vol. 136(1): pp. 1–12 (cit. on p. 81).

LINK JR, EDWIN A: *Trainer for aviators*. US Patent 2,099,857. Nov. 1937 (cit. on p. 45).

LIU, QIANG, JIAFU WAN and KELIANG ZHOU: 'Cloud manufacturing service system for industrial-cluster-oriented application'. *Journal of Internet Technology* (2014), vol. 15(3): pp. 373–380 (cit. on p. 92).

LIU, XINHUA, LI GUO, ZHAOJIE XIA, BONA LU, MINGKUN ZHAO, FANXIAO MENG, ZHOUZHOU LI, JINGHAI LI et al.: 'Harnessing the power of virtual reality'. *Chem Eng Prog* (2012), vol. 108(7): pp. 28–33 (cit. on p. 39).

LIVERANI, ALFREDO, FALKO KUESTER and BERND HAMANN: 'Towards interactive finite element analysis of shell structures in virtual reality'. *Information Visualization, 1999. Proceedings. 1999 IEEE International Conference on*. IEEE. 1999: pp. 340–346 (cit. on p. 39).

LIVESEY, FINBARR: *Defining high value manufacturing*. CBI, 2006 (cit. on p. 15).

LÖDDING, H., A. FRIEDEWALD, M. HEINIG and S. SCHLEUSENER: 'Virtual reality supported assembly planning in the shipbuilding industry'. *Journal of Ship Production* (2011), vol. 27(3): pp. 146–152 (cit. on p. 68).

LOPEZ, RESEARCH LLC: *Building Smarter Manufacturing With The Internet of Things (IoT)*. [Online]. [Accessed 23 October 2017]. 2014. URL: http://cdn.iotwf.com/resources/6/iot%5C_in%5C_manufacturing%5C_january.pdf (cit. on pp. 79, 94).

LUCKE, DOMINIK, CARMEN CONSTANTINESCU and ENGELBERT WESTKÄMPER: 'Smart factory-a step towards the next generation of manufacturing'. *Manufacturing systems and technologies for the new frontier* (2008), vol.: pp. 115–118 (cit. on p. 82).

LUCKEY, PALMER: *Oculus VR*. [Press Release]. [Accessed 23 September 2014]. 2014. URL: http://www.oculus.com/ (cit. on p. 55).

LUDWIG, JOE: 'Second Coming of VR'. [Accessed 25 September 2014]. Ignite Seattle, 2010. URL: https://www.youtube.com/watch?v=9e_8kk4udII (cit. on pp. 50, 52).

LUGRIN, JEAN-LUC, FRED CHARLES, MARC CAVAZZA, MARC LE RENARD, JONATHAN FREEMAN and JANE LESSITER: 'CaveUDK: A VR Game Engine Middleware'. *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*. VRST '12. Toronto, Ontario, Canada, 2012: pp. 137–144 (cit. on p. 53).

MA, SI-QUN and ZHI-LI GAO: 'A solution to integrate CAD/CAE system and VR for complex product design processes'. *Information and Computing Science, 2009. ICIC'09. Second International Conference on*. Vol. 4. IEEE. 2009: pp. 331–333 (cit. on p. 57).

MACEDONIA, MICHAEL R: 'A network software architecture for large scale virtual environments'. PhD thesis. Monterey, California. Naval Postgraduate School, 1995 (cit. on p. 50).

MACEDONIA, MICHAEL R, DONALD P BRUTZMAN, MICHAEL J ZYDA, DAVID R PRATT, PAUL T BARHAM, JOHN FALBY and JOHN LOCKE: 'NPSNET: a multi-player 3D virtual environment over the Internet'. *Proceedings of the 1995 symposium on Interactive 3D graphics*. ACM. 1995: 93–ff (cit. on p. 50).

MACIEL, PAULO WC and PETER SHIRLEY: 'Visual navigation of large environments using textured clusters'. *Proceedings of the 1995 symposium on Interactive 3D graphics*. ACM. 1995: 95–ff (cit. on p. 50).

MAES, PATTIE: 'Artificial life meets entertainment: lifelike autonomous agents'. *Communications of the ACM* (1995), vol. 38(11): pp. 108–114 (cit. on p. 52).

MAGNOX: *Magnox ltd*. [Online]. [Accessed 04 September 2014]. 2014. URL: http://www.magnoxsites.co.uk/ (cit. on p. 7).

MAITEH, BILAL Y, DENIS BLACKMORE, LAYEK ABDEL-MALEK and MING C LEU: 'Swept-volume computation for machining simulation and virtual reality application'. *Journal of Materials Processing and Manufacturing Science* (1999), vol. 7: pp. 380–390 (cit. on p. 60).

MAPLES, CREVE and CRAIG PETERSON: 'μuSE (Multidimensional, User-Oriented Synthetic Environment) A Functionally-Based Human-Computer Interface'. *The International Journal of Virtual Reality* (1995), vol. 1(1): pp. 2–9 (cit. on p. 52).

MARESCAUX, JACQUES, JEAN-MARIE CLÉMENT, VINCENT TASSETTI, CHRISTOPHE KOEHL, STÉPHANE COTIN, YVES RUSSIER, DIDIER MUTTER, HERVÉ DELINGETTE and NICHOLAS AYACHE: 'Virtual reality applied to hepatic surgery simulation: the next revolution.' *Annals of surgery* (1998), vol. 228(5): p. 627 (cit. on p. 39).

MARESCAUX, JACQUES, MICHELLE K SMITH, DANIEL FÖLSCHER, FAEK JAMALI, BENOIT MALASSAGNE and JOEL LEROY: 'Telerobotic laparoscopic cholecystectomy: initial clinical experience with 25 patients'. *Annals of surgery* (2001), vol. 234(1): p. 1 (cit. on p. 51).

MARTIN, HECTOR: *OpenKinect Project Wiki*. [Accessed 17 April 2018]. 2018. URL: https://openkinect.org/wiki/Main_Page (cit. on p. 119).

MARTIN, LOCKHEED: *Collaborative Human Immersive Laboratory (CHIL)*. [Accessed 08 October 2014]. 2014. URL: http://www.lockheedmartin.com/us/products/chil.html (cit. on p. 66).

MAX, NELSON L and BARRY G BECKER: 'Bump shading for volume textures'. *IEEE Computer Graphics and Applications* (1994), vol. 14(4): pp. 18–20 (cit. on p. 54).

MAZURYK, TOMASZ and MICHAEL GERVAUTZ: *Virtual Reality - History, Applications, Technology and Future*. Citeseer, 1996 (cit. on p. 50).

MENDIKOA, I, M SORLI, JI BARBERO, A CARRILLO and A GOROSTIZA: 'Collaborative product design and manufacturing with inventive approaches'. *International Journal of Production Research* (2008), vol. 46(9): pp. 2333–2344 (cit. on p. 82).

MESHLAB: *3D mesh editor*. [Accessed 17 April 2018]. 2018. URL: http://www.meshlab.net/ (cit. on p. 139).

MESSINGER, PAUL R, ELENI STROULIA, KELLY LYONS, MICHAEL BONE, RUN H NIU, KRISTEN SMIRNOV and STEPHEN PERELGUT: 'Virtual worlds—past, present, and future: New directions in social computing'. *Decision Support Systems* (2009), vol. 47(3): pp. 204–228 (cit. on p. 43).

MEYER-ARENDT, JÜRGEN R: 'Introduction to classical and modern optics'. *Englewood Cliffs: Prentice-Hall, 1989, 3rd ed.* (1989), vol. 1 (cit. on p. 229).

MICROSOFT: *HoloLens*. [Accessed 17 April 2018]. 2018. URL: https://www.microsoft.com/en-ca/hololens (cit. on p. 176).

– *Mixed Reality Devices*. [Accessed 17 April 2018]. 2018. URL: https://www.microsoft.com/en-us/store/collections/vrandmixedrealityheadsets (cit. on p. 176).

MILGRAM, PAUL and FUMIO KISHINO: 'A taxonomy of mixed reality visual displays'. *IEICE TRANSACTIONS on Information and Systems* (1994), vol. 77(12): pp. 1321–1329 (cit. on p. 24).

MILGRAM, PAUL, HARUO TAKEMURA, AKIRA UTSUMI and FUMIO KISHINO: 'Augmented reality: A class of displays on the reality-virtuality continuum'. *Photonics for Industrial Applications*. International Society for Optics and Photonics. 1995: pp. 282–292 (cit. on p. 25).

MILNE, AD, DG CHESS, JA JOHNSON and GJW KING: 'Accuracy of an electromagnetic tracking device: a study of the optimal operating range and metal interference'. *Journal of biomechanics* (1996), vol. 29(6): pp. 791–793 (cit. on p. 234).

MIRAGLIOTTA, GIOVANNI, ALESSANDRO PEREGO and ANGELA TUMINO: 'Internet of Things: Smart Present or Smart Future'. *Department of Management, Economics and Industrial Engineering, Politecnico di Milano, Italy* (2012), vol. (cit. on p. 92).

MIRAGLIOTTA, GIOVANNI and FADI SHROUF: 'Using Internet of Things to improve eco-efficiency in manufacturing: a review on available knowledge and a framework for IoT adoption'. *IFIP International Conference on Advances in Production Management Systems*. Springer. 2012: pp. 96–102 (cit. on p. 94).

MIZELL, DAVID W: 'Virtual reality and Augmented Reality in aircraft design and manufacturing'. *Second Annual Symposium on Frontiers of Engineering*. Vol. 31. WA, USA: National Academy of Engineering. 1997 (cit. on p. 75).

MOBACH, MARK P: 'Do virtual worlds create better real worlds?' *Virtual Reality* (2008), vol. 12(3): pp. 163–179 (cit. on p. 52).

MOCHIZUKI, ARITO, TAKASHI AMADA, SAYURI SAWA, TADAYUKI TAKEDA, SHOGO MOTOYASHIKI, KAZUHIRO KOHYAMA, MASATAKA IMURA and KUNIHIRO CHIHARA: 'Fragra: a visual-olfactory VR game'. *ACM SIGGRAPH 2004 Sketches*. ACM. 2004: p. 123 (cit. on p. 44).

MOD: *The future of the United Kingdom's nuclear deterrent: defence white paper 2006 - Fact Sheet 5: The History of the UK's Nuclear Weapons Programme*. White Paper - Fact Sheet 1. [Accessed 04 September 2014]. Cm.6994. London, Stationary Office: Ministry of Defense &

Foreign and Commonwealth Office, 2011. URL: https://www.gov.uk/government/publications/the-future-of-the-united-kingdoms-nuclear-deterrent-defence-white-paper-2006-cm-6994 (cit. on p. 225).

MONAHAN, TERESA, GAVIN MCARDLE and MICHELA BERTOLOTTO: 'Virtual reality for collaborative e-learning'. *Computers & Education* (2008), vol. 50(4): pp. 1339–1353 (cit. on p. 45).

MORI, M: 'Bukimi no tani [The uncanny valley]. Energy, 7 (4) 33-35.(Translated by Karl F. MacDorman and Takashi Minato in 2005 within Appendix B for the paper Androids as an Experimental Apparatus: Why is there an uncanny and can we exploit it?' *Proceedings of the CogSci-2005 Workshop: Toward Social Mechanisms of Android Science*. 1970: pp. 106–118 (cit. on p. 112).

MTBS3D: *Meant to be seen in 3D*. [Online]. [Accessed 26 September 2014]. 2014. URL: http://www.mtbs3d.com/ (cit. on pp. 52, 55).

MÜLLER, WOLFGANG K, ROLF ZIEGLER, ANDRÉ BAUER and EDGAR H SOLDNER: 'Virtual reality in surgical arthroscopic training'. *Computer Aided Surgery* (1995), vol. 1(5): pp. 288–294 (cit. on p. 51).

NAKANISHI, TAKASHI, HISAYO OHTANI, RI-E MIZUOCHI, KUNIO MIYAJI, TAKAO YAMAMOTO, KENJI KOBAYASHI and TETSUJI IMANAKA: 'Residual neutron-induced radionuclides in samples exposed to the nuclear explosion over Hiroshima: comparison of the measured values with the calculated values'. *Journal of radiation research* (1991), vol. 32(Suppl 1): pp. 69–82 (cit. on p. 18).

NAKEVSKA, MARIJA, CASPER VOS, ALEX JUAREZ, JUN HU, GEERT LANGEREIS and MATTHIAS RAUTERBERG: 'Using game engines in mixed reality installations'. *Entertainment Computing-ICEC 2011*. Springer, 2011: pp. 456–459 (cit. on p. 53).

NAM, TEK-JIN and DAVID WRIGHT: 'The development and evaluation of Syco3D: a real-time collaborative 3D {CAD} system'. *Design Studies* (2001), vol. 22(6): pp. 557–582 (cit. on p. 59).

NAMRC: *Industry Intelligence*. [Online]. [Accessed 03 Sep. 2014]. 2014. URL: http://namrc.co.uk/intelligence/uk-new-build-plans/developers/ (cit. on p. 11).

– *Industry Intelligence - Small Modular Reactors*. [Accessed 02 November 2017]. 2017. URL: http://namrc.co.uk/intelligence/smr/ (cit. on p. 12).

NANCE, RICHARD E: *A history of discrete event simulation programming languages*. Technical Report 93-1. Accessed online. Department of Computer Science, Virgina Tech, Virginia 24061: Virginia Polytechnic Institute and State University, June 1993. URL: http://eprints.cs.vt.edu/archive/00000363/01/TR-93-21.pdf (cit. on p. 40).

NAVE, C.R.: *HyperPhysics, Georgia State Universirty,Department of Physics and Astronomy*. [Online]. [Accessed 16 Sep. 2014]. 2012. URL: http://hyperphysics.phy-astr.gsu.edu/hbase/gyr.html (cit. on p. 232).

NCSA: *Bringing Long-Tail Data Into the Light*. [Accessed 08 October 2014]. University of Illinois Urbana-Chapaign, National Center for Supercomputing Applications. 2014. URL: http://browndog.ncsa.illinois.edu/ (cit. on p. 71).

NECHES, ROBERT, RICHARD E FIKES, TIM FININ, THOMAS GRUBER, RAMESH PATIL, TED SENATOR and WILLIAM R SWARTOUT: 'Enabling technology for knowledge sharing'. *AI magazine* (1991), vol. 12(3): p. 36 (cit. on p. 89).

NEUGEBAUER, R., F. PURZEL, A SCHREIBER and T. RIEDEL: 'Virtual reality-aided planning for energy-autonomous factories'. *Industrial Informatics (INDIN), 2011 9th IEEE International Conference on*. July 2011: pp. 250–254 (cit. on pp. 57, 68).

NIA: *Planning our electric future: a White Paper for secure, affordable and low carbon electricity*. [White Paper]. [Accessed 03 September 2014]. CM.8099. London, Stationary Office: Department of Energy & Climate Change, July 2011. URL: http://issuu.com/nuclear_industry_association/docs/nia_capability_2012 (cit. on p. 11).

NIESEN, MELISSA R and GR LUECKE: 'Virtual dynamic prototyping for operator interface design'. *Robot and Human Interaction, 1999. RO-MAN'99. 8th IEEE International Workshop on*. IEEE. 1999: pp. 357–361 (cit. on p. 51).

NIST: *National Institute of Standards and Technology. Workshop report on foundations for innovation in cyber-physical systems*. [Online]. [Accessed 23 October 2017]. 2013. URL: https://www.nist.gov/sites/default/files/documents/el/CPS-WorkshopReport-1-30-13-Final.pdf (cit. on p. 84).

NATIONAL NUCLEAR LABORATORY (NNL): *Small Modular Reactors (SMR) Feasibility Study*. [Accessed 02 November 2017]. 2017. URL: http://namrc.co.uk/wp-content/uploads/2015/01/smr-feasibility-study-december-2014.pdf (cit. on p. 13).

NOE, P and K ZABANEH: 'Relative GPS'. *Position Location and Navigation Symposium, 1994., IEEE*. IEEE. 1994: pp. 586–590 (cit. on p. 229).

NOMACHINE: *NoMachine Remote Desktop*. [Accessed 17 April 2018]. 2018. URL: https://www.nomachine.com/ (cit. on p. 132).

NOVAK-MARCINČIN, JOZEF, MARTIN DOLIAK, SERGEJ HLOCH and TODOR ERGIĆ: 'Application of the virtual reality modelling language to computer aided robot control system ROANS'. *Strojarstvo* (2010), vol. 52(2): pp. 227–232 (cit. on p. 67).

NUGEN: *NuGeneration*. [Online]. [Accessed 03 Sep. 2014]. 2014. URL: http://www.nugeneration.com/ (cit. on p. 11).

NVIDIA: *Quadro Comparison Chart*. [Online]. [Accessed 26 September 2014]. 2014. URL: http://www.nvidia.com/object/compare-quadro-gpus.html (cit. on p. 50).

O'REILLY, J.J.: 'Introduction to SLAM II and SLAMSYSTEM'. *Simulation Conference Proceedings, 1995. Winter*. Dec. 1995: pp. 467–471 (cit. on p. 61).

OCULUS: *Oculus Connect 2014 - Crescent Bay Prototype*. [Press Release]. [Accessed 23 September 2014]. 2014. URL: http://www.oculus.com/blog/oculus-connect-2014/ (cit. on p. 44).

OED: *Oxford English Dictionary*. [Online]. [Accessed 6 Sep. 2014]. 2014. URL: http://www.oed.com (cit. on pp. 24–26, 29, 31, 40).

ONIZAWA, KUNIO, HIROYUKI NISHIKAWA and HIROTO ITOH: 'Development of probabilistic fracture mechanics analysis codes for reactor pressure vessels and piping considering welding residual stress'. *International Journal of Pressure Vessels and Piping* (2010), vol. 87(1): pp. 2–10 (cit. on p. 19).

ONOSATO, MASAHIKO and KAZUAKI IWATA: 'Development of a Virtual Manufacturing System by Integrating Product Models and Factory Models'. *{CIRP} Annals - Manufacturing Technology* (1993), vol. 42(1): pp. 475–478 (cit. on p. 60).

OFFICE OF NUCLEAR REGULATION (ONR): *Design acceptance for the AP1000® reactor*. [Accessed 02 November 2017]. 2017. URL: http://news.onr.org.uk/2017/03/design-acceptance-for-the-ap1000-reactor/ (cit. on p. 10).

ONR: *Japanese earthquake and tsunami: Implications for the UK nuclear industry*. Final Report. [Accessed 06 October 2014]. Office of Nuclear Regulation, 2011. URL: http://www.onr.org.uk/fukushima/final-report.pdf (cit. on p. 9).

ONR: *NEW REACTORS DIVISION - UK Advanced Boiling Water Reactor (ABWR) Regulation Sub-Division*. [Accessed 02 November 2017]. 2017. URL: http://www.onr.org.uk/civil-nuclear-reactors/newydd-wylfa-intervention-strategy.pdf (cit. on p. 10).

ONR: *Office for Nuclear Regulation- A guide to Nuclear Regulation in the UK*. 2016 Update. [Accessed 06 October 2016]. Office of Nuclear Regulation, 2016. URL: http://www.onr.org.uk/documents/a-guide-to-nuclear-regulation-in-the-uk.pdf (cit. on p. 1).

ORIN: *What is ORiN?* [Accessed 03 October 2014]. Japan Robot Association. 2014. URL: http://www.orin.jp/e/?page_id=4 (cit. on p. 37).

OTA, DAVID, BOWEN LOFTIN, TIM SAITO, ROBERT LEA and JAMES KELLER: 'Virtual reality in surgical education'. *Computers in Biology and Medicine* (1995), vol. 25(2): pp. 127–137 (cit. on pp. 65, 73).

OUIJAPAW: *OuijaPaw Games LLC*. [Online]. [Accessed 24 February 2018]. 2018. URL: https://assetstore.unity.com/publishers/7219 (cit. on p. 101).

PAELKE, VOLKER: 'Augmented reality in the smart factory: Supporting workers in an industry 4.0. environment'. *Emerging Technology and Factory Automation (ETFA), 2014 IEEE*. IEEE. 2014: pp. 1–4 (cit. on pp. 82, 97).

PAN, CHUNXIA: 'Integrating CAD files and automatic assembly sequence planning'. (2005), vol. (cit. on pp. 63, 69, 76).

PANETTO, HERVÉ, MICHELE DASSISTI and ANGELA TURSI: 'ONTO-PDM: Product-driven ONTOlogy for Product Data Management interoperability within manufacturing process environment'. *Advanced Engineering Informatics* (2012), vol. 26(2): pp. 334–348 (cit. on p. 88).

PATRICK, EMILEE, DENNIS COSGROVE, ALEKSANDRA SLAVKOVIC, JENNIFER A RODE, THOM VERRATTI and GREG CHISELKO: 'Using a large projection screen as an alternative to head-mounted displays for virtual environments'. *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM. 2000: pp. 478–485 (cit. on p. 30).

PROCESSENGINEERING: *Brave new world: Industry 4.0 technology*. [Online]. [Accessed 23 October 2017]. 2013. URL: http://processengineering.co.uk/article/2017121/brave-new-world-indu (cit. on pp. 79, 94).

PROF. DR. HENNING KAGERMANN Prof. Dr. Wolfgang Wahlster, DR. JOHANNES HELBIG: *Umsetzungsempfehlungen für das Zukunftsprojekt Industrie 4.0*. [White Paper]. [Accessed 23 October 2017]. 2012. URL: http://www.forschungsunion.de/pdf/industrie_4_0_umsetzungsempfehlungen.pdf (cit. on p. 93).

PSOTKA, JOSEPH: 'Immersive training systems: Virtual reality and education and training'. *Instructional science* (1995), vol. 23(5-6): pp. 405–431 (cit. on pp. 9, 65).

QIAO, GUIXIU, ROBERTO F LU and CHARLES MCLEAN: 'Flexible manufacturing systems for mass customisation manufacturing'. *International Journal of Mass Customisation* (2006), vol. 1(2-3): pp. 374–393 (cit. on p. 84).

QIU, MEIKANG and EDWIN HSING-MEAN SHA: 'Energy-aware online algorithm to satisfy sampling rates with guaranteed probability for sensor applications'. *Lecture Notes in Computer Science* (2007), vol. 4782: p. 156 (cit. on p. 92).

QIU, MEIKANG, CHUN XUE, ZILI SHAO, QINGFENG ZHUGE, MEILIN LIU and EDWIN SHA: 'Efficent algorithm of energy minimization for heterogeneous wireless sensor network'. *Embedded and Ubiquitous Computing* (2006), vol.: pp. 25–34 (cit. on p. 92).

QUALITY, AMERICAN SOCIETY of: *ASQ Manufacturing Survey: Outlook Positive, SR Interest Mixed*. [US Survey - Press release]. [Accessed 23 October 2017]. 2014. URL: http://www.prweb.com/releases/2014/12/prweb12396867.htm (cit. on p. 96).

RANASINGHE, N., A. CHEOK, R. NAKATSU and E.Y. DO: 'Simulating the sensation of taste for immersive experiences'. *Proceedings of the 2013 ACM international workshop on Immersive media experiences*. ACM. 2013: pp. 29–34 (cit. on p. 44).

RANASINGHE, N., K. LEE, R. NAKATSU and E.Y. DO: 'FunRasa: an interactive drinking platform'. *Proceedings of the 8th International Conference on Tangible, Embedded and Embodied Interaction*. ACM. 2014: pp. 133–136 (cit. on p. 44).

RASHID, MOHDFADZILFAISAE, WINDO HUTABARAT and ASHUTOSH TIWARI: 'A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches'. English. *The International Journal of Advanced Manufacturing Technology* (2012), vol. 59(1-4): pp. 335–349 (cit. on pp. 39, 63).

RATTS, ERIC B., YI LU MURPHEY and YOUNING ZHOU: 'Thermal modeling of controlled atmosphere brazing process using virtual reality technology'. *Applied Thermal Engineering* (2000), vol. 20(17): pp. 1667–1678 (cit. on p. 39).

REALM: *University of Strathclyde demonstrate augmented and virtual reality construction tool*. [Accessed 02 November 2017]. 2017. URL: http://www.urbanrealm.com/news/6801/University_of_Strathclyde_demonstrate_augmented_and_virtual_reality_construction_tool.html (cit. on p. 58).

REDDISH, S., C. FREEMAN and M. LEWIS: *Apparatus, methods, computer programs, and non-transitory computer readable storage mediums for enabling remote control of one or more devices*. GB Patent GB2549264A. Oct. 2017. URL: https://patents.google.com/patent/GB2549264A/en (cit. on p. c).

– *Apparatus, methods, computer programs, and non-transitory computer readable storage mediums for enabling remote control of one or more devices*. US Patent US20170293275A1. Oct. 2017. URL: https://patents.google.com/patent/US20170293275A1/en (cit. on p. c).

– *Apparatus, methods, computer programs, and non-transitory computer readable storage mediums for enabling remote control of one or more devices*. Chinese Patent CN107422686A. Dec. 2017. URL: https://patents.google.com/patent/CN107422686A/en (cit. on p. c).

REDDISH, STEPHEN: *kinect2_tracker*. [Accessed 17 April 2018]. 2018. URL: https://github.com/mcgi5sr2/kinect2%5C_tracker (cit. on p. 120).

REGAN, MATTHEW and RONALD POSE: 'Priority Rendering with a Virtual Reality Address Recalculation Pipeline'. *Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques*. SIGGRAPH '94. New York, NY, USA, 1994: pp. 155–162 (cit. on p. 50).

REHN, GORDON D, MARCO LEMESSI, JUDY M VANCE and DENIS V DOROZHKIN: 'Integrating operations simulation results with an immersive virtual reality environment'. *Proceedings of the 36th conference on Winter simulation*. Winter Simulation Conference. 2004: pp. 1713–1719 (cit. on p. 63).

REMONDINO, FABIO, ALBERTO GUARNIERI and ANTONIO VETTORE: '3D modeling of close-range objects: photogrammetry or laser scanning?' *Electronic Imaging 2005*. International Society for Optics and Photonics. 2005: pp. 216–225 (cit. on p. 61).

REUTERS: *How two cutting edge U.S. nuclear projects bankrupted Westinghouse*. [Accessed 02 November 2017]. 2017. URL: https://www.reuters.com/article/us-toshiba-accounting-westinghouse-nucle/how-two-cutting-edge-u-s-nuclear-projects-bankrupted-westinghouse-idUSKBN17Y0CQ (cit. on p. 13).

RIEDL, MATTHIAS, HOLGER ZIPPER, MARCO MEIER and CHRISTIAN DIEDRICH: 'Cyber-physical systems alter automation architectures'. *Annual Reviews in Control* (2014), vol. 38(1): pp. 123–133 (cit. on p. 84).

RIETZLER, MICHAEL, FLORIAN GEISELHART, JANEK THOMAS and ENRICO RUKZIO: 'FusionKit: a generic toolkit for skeleton, marker and rigid-body tracking'. *Proceedings of the 8th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*. ACM. 2016: pp. 73–84 (cit. on pp. 121, 134–136, 169).

RITCHIE, JAMES M, THEODORE LIM, RS SUNG, JONATHAN R CORNEY and H REA: 'The Analysis of Design and Manufacturing Tasks Using Haptic and Immersive VR-Some Case Studies''. *Product Engineering*. Springer, 2008: pp. 507–522 (cit. on p. 66).

RÓDENAS, J, I ZARZA, MC BURGOS, A FELIPE and ML SÁNCHEZ-MAYORAL: 'Developing a virtual reality application for training nuclear power plant operators: setting up a database containing dose rates in the refuelling plant'. *Radiation protection dosimetry* (2004), vol. 111(2): pp. 173–180 (cit. on p. 67).

ROLLAND, JANNICK P, LARRY DAVIS and YOHAN BAILLOT: 'A survey of tracking technology for virtual environments'. *Fundamentals of wearable computers and augmented reality* (2001), vol. 1: pp. 67–112 (cit. on pp. 45, 231).

ROLLS-ROYCE: *Rolls-Royce to partner with Chinese reactor vendor CGN to deliver low carbon energy future*. [Press Release]. [Accessed 03 Sep. 2014]. 2014. URL: http://www.rolls-royce.com/news/press_releases/2014/180614_carbon_energy_future.jsp (cit. on p. 10).

ROQUES, FABIEN A, WILLIAM J NUTTALL, DAVID M NEWBERY, RICHARD de NEUFVILLE and STEPHEN CONNORS: 'Nuclear power: a hedge against uncertain gas and carbon prices?' *The Energy Journal* (2006), vol.: pp. 1–23 (cit. on p. 9).

ROTHBAUM, BARBARA OLASOV, LARRY HODGES, PAGE L ANDERSON, LARRY PRICE and SAMANTHA SMITH: 'Twelve-month follow-up of virtual reality and standard exposure therapies for the fear of flying.' *Journal of consulting and Clinical Psychology* (2002), vol. 70(2): p. 428 (cit. on p. 51).

ROWLEY, JENNIFER E: 'The wisdom hierarchy: representations of the DIKW hierarchy'. *Journal of Information Science* (2007), vol. (cit. on pp. 34, 35).

RÜSSMANN, MICHAEL, MARKUS LORENZ, PHILIPP GERBERT, MANUELA WALDNER, JAN JUSTUS, PASCAL ENGEL and MICHAEL HARNISCH: 'Industry 4.0: The future of productivity and growth in manufacturing industries'. *Boston Consulting Group* (2015), vol. 9 (cit. on pp. 97, 181).

SALBU, B, T KREKLING, DH OUGHTON, G ØSTBY, VA KASHPAROV, TL BRAND and JP DAY: 'Hot particles in accidental releases from Chernobyl and Windscale nuclear installations'. *Analyst* (1994), vol. 119(1): pp. 125–130 (cit. on p. 18).

SAVAGE-CARMONA, JESUS, MARK BILLINGHURST and ALISTAIR HOLDEN: 'The VirBot: a virtual reality robot driven with multimodal commands'. *Expert Systems with Applications* (1998), vol. 15(3–4): pp. 413–419 (cit. on p. 45).

SCHNABEL, MARC AUREL, XIANGYU WANG, HARTMUT SEICHTER and TOM KVAN: 'From virtuality to reality and back'. *Proceedings of the International Association of Societies of Design Research* (2007), vol. 1: p. 15 (cit. on p. 24).

SCHÖFFEL, FRANK: 'Online radiosity in interactive virtual reality applications'. *Proceedings of the ACM symposium on Virtual reality software and technology*. ACM. 1997: pp. 201–208 (cit. on p. 51).

SCHÖN, DONALD A: *The reflective practitioner: How professionals think in action*. Vol. 5126. Basic books, 1983 (cit. on p. 58).

SCHRÖDINGER: *A molecular visualization system on an open source foundation, maintained and distributed by Schrödinger*. [Accessed 03 October 2014]. Schrödinger. 2014. URL: http://www.pymol.org/ (cit. on p. 38).

SCHUEMIE, MARTIJN J, PETER VAN DER STRAATEN, MEREL KRIJN and CHARLES APG VAN DER MAST: 'Research on presence in virtual reality: A survey'. *CyberPsychology & Behavior* (2001), vol. 4(2): pp. 183–201 (cit. on p. 56).

SCHULTZ, KENNETH L, JOHN O MCCLAIN and L.JOSEPH THOMAS: 'Overcoming the dark side of worker flexibility'. *Journal of Operations Management* (2003), vol. 21(1): pp. 81–92 (cit. on pp. 15, 16).

SCOTT, HAROLD A: 'Modeling aircraft assembly operations'. *Proceedings of the 26th conference on Winter simulation*. Society for Computer Simulation International. 1994: pp. 920–927 (cit. on pp. 39, 61).

SCOTT, PETER M, MARIE-CHRISTINE MEUNIER, DENIS DEYDIER, SARAH SILVESTRE, ALAIN TRENTY and RD KANE: 'An analysis of baffle/former bolt cracking in French PWRs'. *ASTM STP* (2000), vol. 1401: pp. 210–223 (cit. on p. 15).

SCOTXW: *The input/output loop: Human Machine Interface (HMI)*. [Accessed 03 October 2014]. Wikipedia. 2014. URL: http://commons.wikimedia.org/wiki/User:ScotXW (cit. on p. 150).

SETH, ABHISHEK, JUDYM. VANCE and JAMESH. OLIVER: 'Virtual reality for assembly methods prototyping: a review'. *Virtual Reality* (2011), vol. 15(1): pp. 5–20 (cit. on p. 65).

SHAHMANESH, NARGESS: 'Computer aided design and computer-assisted production'. *Automotive engineer* (1998), vol. 23(11): pp. 44–46 (cit. on p. 57).

SHARP, NEIL: 'Production Engineer'. [per comm]. "Discussion on Nuclear Manufacture". Sept. 2014 (cit. on p. 18).

SHAW, CHRIS and MARK GREEN: 'The MR toolkit peers package and experiment'. *Virtual Reality Annual International Symposium, 1993., 1993 IEEE*. IEEE. 1993: pp. 463–469 (cit. on p. 50).

SHAW, CHRIS, MARK GREEN, JIANDONG LIANG and YUNQI SUN: 'Decoupled simulation in virtual reality with the MR toolkit'. *ACM Transactions on Information Systems (TOIS)* (1993), vol. 11(3): pp. 287–317 (cit. on p. 50).

SHERMAN, WILLIAM R and ALAN B CRAIG: 'Understanding Virtual Reality—Interface, Application, and Design'. *Presence: Teleoperators and Virtual Environments* (2003), vol. 12(4): pp. 441–442 (cit. on p. 43).

– *Understanding virtual reality: Interface, application, and design*. Elsevier, 2002 (cit. on p. 48).

SHI, JIANHUA, JIAFU WAN, HEHUA YAN and HUI SUO: 'A survey of cyber-physical systems'. *Wireless Communications and Signal Processing (WCSP), 2011 International Conference on*. IEEE. 2011: pp. 1–6 (cit. on p. 84).

SHIRATUDDIN, MOHD FAIRUZ and WALID THABET: 'Virtual office walkthrough using a 3D game engine'. *International Journal of Design Computing* (2002), vol. 4 (cit. on p. 52).

SHROUF, FADI, JOAQUIN ORDIERES-MERE, ALVARO GARCIA-SANCHEZ and MIGUEL ORTEGA-MIER: 'Optimizing the production scheduling of a single machine to minimize total energy consumption costs'. *Journal of Cleaner Production* (2014), vol. 67: pp. 197–207 (cit. on p. 94).

SHROUF, FADI, JOAQUIN ORDIERES and GIOVANNI MIRAGLIOTTA: 'Smart factories in Industry 4.0: A review of the concept and of energy management approached in production based on the Internet of Things paradigm'. *Industrial Engineering and Engineering Management (IEEM), 2014 IEEE International Conference on*. IEEE. 2014: pp. 697–701 (cit. on p. 92).

SIEMENS: *Jack and Process Simulate Human*. [Press Release]. [Accessed 29 September 2014]. 2014. URL: http://www.plm.automation.siemens.com/en_us/products/tecnomatix/manufacturing-simulation/human-ergonomics/jack.shtml (cit. on p. 66).

– *TechnoMatix for Plant Simulation*. [Accessed 30 September 2014]. 2014. URL: http://www.plm.automation.siemens.com/en_us/products/tecnomatix/manufacturing-simulation/material-flow/plant-simulation.shtml (cit. on pp. 61, 63).

SKODA: *Equipment for PWR and BWR Type of Nuclear Power Plants*. [Accessed 06 October 2014]. Skoda JS a.s. 2014. URL: http://www.skoda-js.cz/en/products-and-services/equipment-for-the-type-pwr-and-bwr-nuclear-power-plants/index.shtml (cit. on p. 14).

SLATER, MEL: 'Place illusion and plausibility can lead to realistic behaviour in immersive virtual environments'. *Philosophical Transactions of the Royal Society B: Biological Sciences* (2009), vol. 364(1535): pp. 3549–3557 (cit. on pp. 30, 31, 44).

SLATER, MEL, MARTIN USOH and ANTHONY STEED: 'Taking steps: the influence of a walking technique on presence in virtual reality'. *ACM Transactions on Computer-Human Interaction (TOCHI)* (1995), vol. 2(3): pp. 201–219 (cit. on p. 227).

SMITHSONIAN: *Oil Spill Gulf of Mexico 2010*. [Accessed 06 October 2014]. Smithsonian National Museum of Natural History. 2014. URL: http://ocean.si.edu/gulf-oil-spill (cit. on p. 18).

SNOWDON, D, A WEST and TOBY LJ HOWARD: 'Towards the next generation of Human-Computer Interface'. *Informatique'93: Interface to Real & Virtual Worlds* (1993), vol.: pp. 399–408 (cit. on p. 50).

SNOWDON, DAVID N and ADRIAN J WEST: 'AVIARY- Design issues for future large-scale virtual environments'. *Presence- Teleoperators and Virtual Environments* (1994), vol. 3(4): pp. 288–308 (cit. on p. 50).

SO, RICHARD HY and MICHAEL J GRIFFIN: 'Compensating lags in head-coupled displays using head position prediction and image deflection'. *Journal of Aircraft* (1992), vol. 29(6): pp. 1064–1068 (cit. on p. 50).

SOUZA, MARIELLACONSONIFLORENZANO, MARCO SACCO and ARTHURJOSÉVIEIRA PORTO: 'Virtual manufacturing as a way for the factory of the future'. *Journal of Intelligent Manufacturing* (2006), vol. 17(6): pp. 725–735 (cit. on p. 60).

SRINIVASAN, M.A. and R. ZIMMER: 'Machine Haptics'. *Encyclopedia of Neuroscience*. Ed. by SQUIRE, LARRY R. Oxford: Academic Press, 2009: pp. 589–595 (cit. on p. 45).

SRINIVASAN, MANDAYAM A and CAGATAY BASDOGAN: 'Haptics in virtual environments: Taxonomy, research status, and challenges'. *Computers & Graphics* (1997), vol. 21(4): pp. 393–404 (cit. on pp. 32, 45).

STATE COUNCIL OF CHINA, PREMIER of the and K. Q. LI: *Report on the work of the government," in Proceedings of the 3rd Session of the 12th National People's Congress, March 2015*. [Online]. [Accessed 23 October 2017]. 2017. URL: http://english.gov.cn/archive/publications/2015/03/05/content_281475066179954.htm (cit. on p. 96).

STEREOSCOPY.COM: *What is Stereoscopy*. [Online]. Accessed 6 Sep. 2014]. 2014. URL: http://www.Stereoscopy.com (cit. on p. 26).

STRASSBURGER, STEFFEN, THOMAS SCHULZE, MARCO LEMESSI and GORDON D. REHN: 'Temporally Parallel Coupling of Discrete Simulation Systems with Virtual Reality Systems'. *Proceedings of the 37th Conference on Winter Simulation*. WSC '05. Orlando, Florida: Winter Simulation Conference, 2005: pp. 1949–1957 (cit. on pp. 62, 63, 74).

STURMAN, DAVID J and DAVID ZELTZER: 'A survey of glove-based input'. *Computer Graphics and Applications, IEEE* (1994), vol. 14(1): pp. 30–39 (cit. on p. 48).

SUNG, RAYMOND CW, JAMES M RITCHIE, THEODORE LIM and HUGO MEDELLIN: 'Assembly planning and motion study using virtual reality'. *ASME-AFM 2009 World Conference on Innovative Virtual Reality*. American Society of Mechanical Engineers. 2009: pp. 31–38 (cit. on p. 65).

SUTHERLAND, IVAN E: 'A head-mounted three dimensional display'. *Proceedings of the December 9-11, 1968, fall joint computer conference, part I*. ACM. 1968: pp. 757–764 (cit. on pp. 48, 233).

– 'The Ultimate Display'. *Proceedings of IFIP Congress 2*. 1965: pp. 506–509 (cit. on p. 46).

SUTHERLAND, IVAN E and GARY W HODGMAN: *Computer Graphics Clipping System For Polygons*. US Patent 3,816,726. June 1974 (cit. on p. 48).

SYGNALOW, ZESPOL PRZETWARZANIA: *Audio and video raytracer - RAYAV*. [Online]. [Accessed 23 September 2014]. 2014. URL: http://www.dsp.agh.edu.pl/en:research:rayav%5C#.VCFXTK0gtVZ (cit. on p. 44).

SZTIPANOVITS, JANOS, XENOFON KOUTSOUKOS, GABOR KARSAI, NICHOLAS KOTTENSTETTE, PANOS ANTSAKLIS, VIJAY GUPTA, BILL GOODWINE, JOHN BARAS and SHIGE WANG: 'Toward a science of cyber–physical system integration'. *Proceedings of the IEEE* (2012), vol. 100(1): pp. 29–44 (cit. on p. 87).

TAMURA, HIDEYUKI, HIROYUKI YAMAMOTO and AKIHIRO KATAYAMA: 'Mixed reality: Future dreams seen at the border between real and virtual worlds'. *Computer Graphics and Applications, IEEE* (2001), vol. 21(6): pp. 64–70 (cit. on p. 59).

TAO, FEI, LIN ZHANG and AYC NEE: 'A review of the application of grid technology in manufacturing'. *International Journal of Production Research* (2011), vol. 49(13): pp. 4119–4155 (cit. on p. 83).

TAO, FEI, YING ZUO, LI DA XU and LIN ZHANG: 'IoT-based intelligent perception and access of manufacturing resource toward cloud manufacturing'. *IEEE Transactions on Industrial Informatics* (2014), vol. 10(2): pp. 1547–1557 (cit. on p. 92).

TARR, MICHAEL J and WILLIAM H WARREN: 'Virtual reality in behavioral neuroscience and beyond'. *nature neuroscience* (2002), vol. 5: pp. 1089–1092 (cit. on p. 45).

TECHNOLOGIES, UNITY: *Unity3D Game Engine*. [Accessed 01 October 2014]. 2014. URL: http://unity3d.com/ (cit. on pp. 53, 99, 101).

TECNATOM, SA: 'Using virtual reality for the training of the Metallographic Replica technique used to inspect power plants by TECNATOM SA'. *Joint VR Conference of euroVR and EGVE*. 2011: p. 53 (cit. on p. 67).

TEUNISSEN, CLEON: *The Sagnac Effect*. [Online]. [Accessed 12 Sep. 2014]. 2010. URL: http://www.cleonis.nl/physics/phys256/sagnac.php (cit. on p. 230).

TEXAS-INSTRUMENTS, TI: *IoT made easy*. [Online]. [Accessed 24 February 2018]. 2018. URL: http://www.ti.com/ww/en/wireless_connectivity/sensortag/ (cit. on p. 100).

THESTONEFOX: *VRTK*. [Accessed 17 April 2018]. 2018. URL: https://github.com/thestonefox/VRTK (cit. on p. 146).

TOYOTA: *Toyota Production System*. [Accessed 17 April 2018]. 2018. URL: http://www.toyota-global.com/company/vision_philosophy/toyota_production_system/ (cit. on p. 160).

TSENG, H-E: 'Guided genetic algorithms for solving a larger constraint assembly problem'. *International Journal of Production Research* (2006), vol. 44(3): pp. 601–625 (cit. on p. 66).

TUEGEL, ERIC J, ANTHONY R INGRAFFEA, THOMAS G EASON and S MICHAEL SPOTTSWOOD: 'Reengineering aircraft structural life prediction using a Digital Twin'. *International Journal of Aerospace Engineering* (2011), vol. 2011 (cit. on p. 91).

UKAEA: *UK Atomic Energy Authority Evolution of the UK Atomic Energy Authority*. [Press Release]. [Accessed 04 September 2014]. 2014. URL: https://www.gov.uk/government/organisations/uk-atomic-energy-authority/about (cit. on p. 7).

UKERC: *British public split on nuclear power*. [Accessed 06 October 2014]. UK Energy Research Centre. 2014. URL: http://www.ukerc.ac.uk/support/article3253-British-public-split-on-nuclear-power (cit. on p. 9).

ULUPINAR, F. and R. NEVATIA: 'Shape from contour: straight homogeneous generalized cylinders and constant cross section generalized cylinders'. *Pattern Analysis and Machine Intelligence, IEEE Transactions on* (Feb. 1995), vol. 17(2): pp. 120–135 (cit. on p. 60).

VALVE-CORPORATION: *Steam VR Plugin for Unity3D*. [Accessed 17 April 2018]. 2018. URL: https:

//assetstore.unity.com/packages/templates/systems/steamvr-plugin-32647 (cit. on pp. 144, 168).

Van Dam, Andries, Andrew Forsberg, David H Laidlaw, Joseph J LaViola and Rosemary M Simpson: 'Immersive VR for scientific visualization: A progress report'. *Computer Graphics and Applications, IEEE* (2000), vol. 20(6): pp. 26–52 (cit. on p. 25).

Van Krevelen, DWF and R Poelman: 'A survey of augmented reality technologies, applications and limitations'. *International Journal of Virtual Reality* (2010), vol. 9(2): p. 1 (cit. on p. 9).

Van Sickle, Jan: *GPS for land surveyors*. CRC Press, 2001 (cit. on p. 229).

Ventä, Olli and Björn Wahlström: 'Investigating the case of Open Source applications within nuclear power'. *Enlarged Halden Programme Group meeting, Storefjell, Norway*. 2007: pp. 12–15 (cit. on p. 119).

VESTAS: *Virtual Reality and VESTAS*. [Accessed 02 November 2015]. 2014. URL: http://www.vestaarchitecture.co.uk/virtual-reality/ (cit. on p. 76).

VETL: *Virtual Environment Technology Lab*. [Online]. [Accessed 26 September 2014]. 1997. URL: http://www.crpc.rice.edu/AnnualMeeting/1997/VETL.html (cit. on p. 51).

Vikhorev, Konstantin, Richard Greenough and Neil Brown: 'An advanced energy management framework to promote energy awareness'. *Journal of Cleaner Production* (2013), vol. 43: pp. 103–112 (cit. on p. 94).

Vince, John: *Introduction to virtual reality*. Springer, 2004: p. 36 (cit. on p. 59).

Virtalis: *Case Study - BAE Systems Submarine Solutions*. [Accessed 02 November 2015]. 2014. URL: http://www.virtalis.com/blogs/casestudies/bae-systems-submarine-solutions-3/ (cit. on p. 75).

– *Case Study - Case New Holland*. [Accessed 02 November 2015]. 2014. URL: http://www.virtalis.com/blogs/casestudies/case-new-holland-5/ (cit. on p. 75).

VIRTSim: *Law Enforcement*. [Accessed 01 October 2014]. 2014. URL: http://www.motionreality.com/law-enforcement (cit. on p. 55).

– *VirtSim Military*. [Accessed 01 October 2014]. 2014. URL: http://www.motionreality.com/virtsim-military (cit. on p. 55).

Vlastos, Gregory: 'The philosophy of Socrates: A collection of critical essays'. (1980), vol. (cit. on p. 90).

Vozenilek, John, J Stephen Huff, Martin Reznek and James A Gordon: 'See one, do one, teach one: advanced technology in medical education'. *Academic Emergency Medicine* (2004), vol. 11(11): pp. 1149–1154 (cit. on p. 55).

Wahlster, Wolfgang: 'Semantic technologies for mass customization'. *Towards the Internet of Services: The THESEUS Research Program*. Springer, 2014: pp. 3–13 (cit. on p. 87).

Walter, Bryan E, Jared S Knutzon, Adrian V Sannier and James H Oliver: 'Virtual UAV ground control station'. *AIAA 3rd "Unmanned Unlimited" Technical Conference, Workshop and Exhibit*. 2004 (cit. on p. 55).

Wan, Jia-fu, Di Li, He-hua Yan and Ping Zhang: 'Fuzzy feedback scheduling algorithm based on central processing unit utilization for a software-based computer numerical control system'. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* (2010), vol. 224(7): pp. 1133–1143 (cit. on p. 92).

WAN, JIAFU, HEHUA YAN, QIANG LIU, KELIANG ZHOU, RONGSHUANG LU and DI LI: 'Enabling cyber–physical systems with machine–to–machine technologies'. *International Journal of Ad Hoc and Ubiquitous Computing* (2013), vol. 13(3-4): pp. 187–196 (cit. on p. 84).

WAN, JIAFU, DAQIANG ZHANG, YANTAO SUN, KAI LIN, CAIFENG ZOU and HU CAI: 'VCMIA: a novel architecture for integrating vehicular cyber-physical systems and mobile cloud computing'. *Mobile Networks and Applications* (2014), vol. 19(2): pp. 153–160 (cit. on p. 92).

WANG, G GARY: 'Definition and review of virtual prototyping'. *Journal of Computing and Information Science in engineering* (2002), vol. 2(3): pp. 232–236 (cit. on p. 57).

WANG, QING-HUI, JING-RONG LI, BAO-LI WU and XIAO-MING ZHANG: 'Live parametric design modifications in CAD-linked virtual environment'. English. *The International Journal of Advanced Manufacturing Technology* (2010), vol. 50(9-12): pp. 859–869 (cit. on pp. 58, 59, 69).

WANG, SHIYONG, JIAFU WAN, DI LI and CHUNHUA ZHANG: 'Implementing smart factory of industrie 4.0: an outlook'. *International Journal of Distributed Sensor Networks* (2016), vol. 12(1): p. 3159805 (cit. on p. 95).

WANG, ZHENBIAO, Y. SHEN, S. K. ONG and AY.-C. NEE: 'Assembly Design and Evaluation Based on Bare-Hand Interaction in an Augmented Reality Environment'. *CyberWorlds, 2009. CW '09. International Conference on*. Sept. 2009: pp. 21–28 (cit. on p. 63).

WATERWORTH, JOHN A and EVA L WATERWORTH: 'The meaning of presence'. *Presence-Connect 3* (2003), vol. [Accessed 23 September 2014]. URL: http://www8.informatik.umu.se/jwworth/PRESENCE-meaning.htm (cit. on p. 44).

WEBBINK, R.F. and S.J. HU: 'Automated generation of assembly system-design solutions'. *Automation Science and Engineering, IEEE Transactions on* (Jan. 2005), vol. 2(1): pp. 32–39 (cit. on p. 64).

WEI, KWANG SU, ALAIN CHONG YEE LOONG, YING-MEI LEONG and KENG-BOON OOI: 'Measuring ERP system success: a respecification of the Delone and McLean's IS success model'. *Symposium on progress in information & communication technology*. Citeseer. 2009: pp. 7–8 (cit. on p. 94).

WESTINGHOUSE: *AP1000*. [Online]. [Accessed 04 September 2014]. 2014. URL: https://www.ukap1000application.com/ (cit. on p. 10).

WHEATSTONE, CHARLES: 'Contributions to the Physiology of Vision.-Part the First. On Some Remarkable, and Hitherto Unobserved, Phenomena of Binocular Vision'. English. *Philosophical Transactions of the Royal Society of London* (1838), vol. 128: pp. 371–394 (cit. on p. 26).

WHEELER, JA: 'The participatory anthropic principle'. *The Nature of Scientific Discovery* (1975), vol.: pp. 261–96 (cit. on p. 43).

WHYTE, JENNIFER: *Industrial applications of virtual reality in architecture and construction*. 2003 (cit. on p. 51).

WINKELBACH, S. and F.M. WAHL: 'Shape from 2D Edge Gradients'. *Pattern Recognition*. Ed. by RADIG, BERND and STEFAN FLORCZYK. Vol. 2191. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2001: pp. 377–384 (cit. on p. 60).

WINTLE, TOM: *Alarm sounded over delays to develop UK mini nuclear reactors, theGUARDIAN.com.* [Accessed 02 November 2017]. 2017. URL: https://www.theguardian.com/business/2017/may/02/alarm-sounded-over-delays-to-develop-uk-mini-nuclear-reactors (cit. on p. 13).

WITCHALLS, CLINT and J CHAMBERS: 'The internet of things business index: A quiet revolution gathers pace'. *The Economist Intelligence Unit. Retrieved from http://www. economistinsights. com/analysis/internet-things-business-index* (2013), vol. (cit. on pp. 79, 96).

WORLD NUCLEAR ASSOCIATION (WNA): *Nuclear Development in the United Kingdom*. [Online]. [Accessed 04 September 2014]. 2013. URL: http://www.world-nuclear.org/info/country-profiles/countries-t-z/appendices/nuclear-development-in-the-united-kingdom/ (cit. on pp. 8, 225).

– *Nuclear Power in the United Kingdom*. [Online]. [Accessed 04 September 2017. 2017. URL: http://www.world-nuclear.org/information-library/country-profiles/countries-t-z/united-kingdom.aspx (cit. on p. 8).

WOGMAN, NED A, LEONARD J BOND, ALAN E WALTAR and RE LEBER: 'The nuclear education and staffing challenge: Rebuilding critical skills in nuclear science and technology'. *Journal of Radioanalytical and Nuclear Chemistry* (2005), vol. 263(1): pp. 137–143 (cit. on p. 9).

WOMACK, JAMES P, DANIEL T JONES and DANIEL ROOS: 'The Machine That Changed the World: The Story of Lean Production: How Japan's Secret Weapon in the Global Auto Wars Will Revolutionize Western Industry'. *New York, NY: Rawson Associates* (1990), vol. (cit. on p. 81).

XU, XG, TC CHAO and A BOZKURT: 'VIP-Man: an image-based whole-body adult male model constructed from color photographs of the Visible Human Project for multi-particle Monte Carlo calculations'. *Health Physics* (2000), vol. 78(5): pp. 476–486 (cit. on p. 39).

XU, XUN: 'From cloud computing to cloud manufacturing'. *Robotics and computer-integrated manufacturing* (2012), vol. 28(1): pp. 75–86 (cit. on p. 92).

YAMADA, TOMOYA, SATOSHI YOKOYAMA, TOMOHIRO TANIKAWA, KOICHI HIROTA and MICHITAKA HIROSE: 'Wearable olfactory display: Using odor in outdoor environment'. *Virtual Reality Conference, 2006*. IEEE. 2006: pp. 199–206 (cit. on p. 44).

YAN, HS and CG XUE: 'Decision-making in self-reconfiguration of a knowledgeable manufacturing system'. *International journal of production research* (2007), vol. 45(12): pp. 2735–2758 (cit. on p. 84).

YANG, RUN DANG, XIUMIN FAN, DIANLIANG WU and JUANQI YAN: 'Virtual assembly technologies based on constraint and DOF analysis'. *Robotics and Computer-Integrated Manufacturing* (2007), vol. 23(4): pp. 447–456 (cit. on p. 64).

YAO, Y.X., P.J. XIA, J.S. LIU and J.G. LI: 'A pragmatic system to support interactive assembly planning and training in an immersive virtual environment (I-VAPTS)'. *The International Journal of Advanced Manufacturing Technology* (2006), vol. 30(9-10): pp. 959–967 (cit. on pp. 65, 66).

YASUNARI, TEPPEI J, ANDREAS STOHL, RYUGO S HAYANO, JOHN F BURKHART, SABINE ECKHARDT and TETSUZO YASUNARI: 'Cesium-137 deposition and contamination of Japanese soils due to the Fukushima nuclear accident'. *Proceedings of the National Academy of Sciences* (2011), vol. 108(49): pp. 19530–19534 (cit. on p. 18).

YE, NONG, PRASHANT BANERJEE, AMARNATH BANERJEE and FRED DECH: 'A comparative study of assembly planning in traditional and virtual environments'. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* (1999), vol. 29(4): pp. 546–555 (cit. on p. 65).

YOSHIA, SHUHEI: *Shuhei Yoshida on the Project Morpheus Virtual Reality Headset*. [Interview]. [Accessed 24 September 2014]. 2014. URL: http://www.theverge.com/a/virtual-reality/qa_sony (cit. on p. 71).

Yoshiki, Kaori, H. Saito and M. Mochimaru: 'Reconstruction of 3D Face Model from Single Shading Image Based on Anatomical Database'. *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*. Vol. 4. 2006: pp. 350–353 (cit. on p. 60).

Zha, X.F., S.Y.E. Lim and S.C. Fok: 'Integrated intelligent design and assembly planning: A survey'. English. *The International Journal of Advanced Manufacturing Technology* (1998), vol. 14(9): pp. 664–685. URL: `http://dx.doi.org/10.1007/BF01192287` (cit. on p. 63).

Zhai, Ruifang and Chengda Lin: '3D model generation of complex objects from multiple range images'. *Electric Information and Control Engineering (ICEICE), 2011 International Conference on*. IEEE. 2011: pp. 1–4 (cit. on p. 60).

Zhai, Wenbin, Xiumin Fan, Juanqi Yan and Pengsheng Zhu: 'An integrated simulation method to support virtual factory engineering'. *International Journal of CAD/CAM* (2009), vol. 2(1) (cit. on p. 62).

ZHANG, YU, FEI XIE, YUNWEI DONG, GANG YANG and XINGSHE ZHOU: 'High fidelity virtualization of cyber-physical systems'. *International Journal of Modeling, Simulation, and Scientific Computing* (2013), vol. 4(02): p. 1340005 (cit. on p. 85).

Zheng, Yi, Ru-xin Ning, Jian-hua Liu and Long Du: 'Survey on key techniques of virtual assembly'. *Journal of System Simulation* (2006), vol. 3: p. 031 (cit. on p. 64).

Zhou, Jiehan and Rose Dieng-Kuntz: 'Manufacturing ontology analysis and design: towards excellent manufacturing'. *Industrial Informatics, 2004. INDIN'04. 2004 2nd IEEE International Conference on*. IEEE. 2004: pp. 39–45 (cit. on pp. 88, 90).

Zhu, H., D. Wu and X. Fan: 'Assembly semantics modeling for assembling process planning in virtual environment'. *Assembly Automation* (2010), vol. 30(3): pp. 257–267 (cit. on p. 65).

Zhu, Haiping, Fanmao Liu, Xinyu Shao and Guojun Zhang: 'Integration of rough set and neural network ensemble to predict the configuration performance of a modular product family'. *International Journal of Production Research* (2010), vol. 48(24): pp. 7371–7393 (cit. on p. 84).

Zimmerman, Thomas G, Jaron Lanier, Chuck Blanchard, Steve Bryson and Young Harvill: 'A hand gesture interface device'. *ACM SIGCHI Bulletin*. Vol. 18. ACM. 1987: pp. 189–192 (cit. on pp. 39, 49).

Zuehlke, Detlef: 'SmartFactory—Towards a factory-of-things'. *Annual Reviews in Control* (2010), vol. 34(1): pp. 129–138 (cit. on p. 94).

Zyda, Michael J, David R Pratt, James G Monahan and Kalin P Wilson: 'NPSNET: Constructing a 3D virtual world'. *Proceedings of the 1992 symposium on Interactive 3D graphics*. ACM. 1992: pp. 147–156 (cit. on p. 50).

Acronyms

Notation	**Description**
ABWR	Advanced Boiling Water Reactor
AGR	Advanced Gas Reactor
ALARA	As Low As Reasonably Acceptable
ALARP	As Low As Reasonably Possible

Notation	Description
AMR	Advanced Modular Reactor
AMRC	Advanced Manufacturing Research Centre
API	Application Programming Interface
AR	Augmented Reality
ASCII	American Standard Code for Information Interchange
ASME	American Society of Mechanical Engineers
AWS	Amazon Web Service
B2B	Business to Business
BGS	British Geological Survey
BIM	Building Information Modelling
BOOM	Binocular Omni-Orientated Monitor
CAAP	Computer Assisted Assembly Planning
CAD	Computer Aided Design
CAVE	Computer Assisted Virtual Environment
CFD	Computational Fluid Dynamics
CG	Computer Generated
CGN	China General Nuclear
CNNC	China National Nuclear Corporation
CPPS	Cyber Physical Production System
CPS	Cyber Physical System
CRT	Cathode Ray Tube
DAA	Digitally Assisted Assembly
DAAC	Digitally Assisted Assembly Cell
DAC	Design Augmented by Computer
DBEIS	Department for Business, Energy & Industrial Strategy

Notation	Description
DES	Discrete Event Simulation
DoD	Department of Defence
DOE	Department of Energy
DoF	Degrees of Freedom
DPI	Dye Penetrant Inspection
DRMSE	Distributed Real Manufacturing Simulation Environment
DSE	Distributed Simulation Environment
EAC	Environmentally Assisted Cracking
EDF	Energy De France
EngD	Engineering Doctorate
EoL	End of Life
EPA	Environmental Protection Agency
EPR	Evolutionary Power Reactor
ERP	Enterprise Resource Planning
FEA	Finite Element Analysis
FOSS	Free and Open Source Software
FoV	Field of View
FT	Financial Times
GDA	Generic Design Assessments
GNS	General Nuclear System Limited
GPS	Global Positioning Satellite
GUI	Graphical User Interface
HAL	Hardware Abstraction Layer
HCI	Human Computer Interaction
HMD	Head Mounted Display

Notation	Description
HS&E	Health Safety & Environment
HUD	Head Up Display
ICS	Instrument & Control Systems
ICT	Information & Communications Technology
IoF	Incredibility of failure
IoT	Internet of Things
IP	Internet Protocol address
IR4.0	4^{th} Industrial Revolution
IVR	Immersive Virtual Reality
JET	Joint European Torus
JIT	Just In Time
JSF	Joint Strike Fighter
JSON	JavaScript Object Notation
KAERI	Korea Atomic Energy Research Institute
LEEP	Large Expanse Enhanced Perspective
LH	Left Hand
LoD	Level of Detail
MES	Manufacturing Executions System
MMO	Massively Multi-player Online game
MoU	Memorandum of Understanding
MR	Mixed Reality
MRP	Material Resource Planning
MSR	Molten Salt Reactor
NDE	Non-Destructive Evaluation
NDT	Non-Destructive Testing

Notation	Description
NIA	Nuclear Industry Association
NNL	National Nuclear Laboratory
NPP	Nuclear Power Plant
NPS	National Policy Statement
OEE	Overall Equipment Effectiveness
ONR	Office of Nuclear Regulation
PDM	Product Data Management
PDMS	Product Data Management System
PhD	Doctorate of Philosophy
PI	Place Illusion
PLM	Product Life-cycle Management
PP&C	Production Planning & Control
PPE	Personal Protective Equipment
PsI	Plausibility Illusion
PTSD	Post Traumatic Stress Disorder
PWR	Pressurised Water Reactor
QA	Quality Assurance
RDP	Remote Desktop Protocol
RE	Research Engineer
RFID	Radio-Frequency Identification
RH	Right Hand
ROS	Robot Operating System
RR	Rolls-Royce
SDK	Software Development Kit
SME	Small to Medium Enterprise

Notation	Description
SMR	Small Modular Reactor
ToF	Time of Flight
TRL	Technology Readiness Level
UKAEA	United Kingdom Atomic Energy Authority
UNC	University of North Carolina
VAPP	Virtual Assembly Process Planning
VAT	Virtual Assembly Training
VCASS	Coupled Airbourne Systems Simulator
VE	Virtual Environment
VIEW	Virtual Interface Environment Workstation
VIVED	Virtual Visual Environment Display
VM	Virtual Manufacturing
VoIP	Voice over Internet Protocol
VR	Virtual Reality
VRTK	Virtual Reality Toolkit
WNA	World Nuclear Association

APPENDICES A

Background Appendix

A.1 UK Pre-Civil Nuclear History

The nuclear industry in the UK began in 1941 when then prime minister Winston Churchill started research into creating an atomic weapon during the second world war. However due to a lack of funding, access to sufficient materials the program was slow moving. This lead to an agreement between prime minister Winston Churchill and U.S president Roosevelt that the British program should be incorporated into the American "Manhattan Project". Post war, the 1946 American 'McMahon Act' effectively banned foreign involvement in the U.S nuclear weapons program. This lead to prime minister Atlee resuming a UK based program to create their own atomic weapons (MoD, 2011).

The first reactors built in the UK were to support the revived nuclear weapons program. The work undertaken at Harwell in Oxfordshire led to the creation of the Atomic Energy Research Establishment (AERE), led to creation of the Graphite Low Energy Pile (GLEEP), followed closely by the British Experimental Pile '0' (BEPO). The two reactors were used as test beds for the technology that was to be used in the Windscale Piles, one of which still stands on the modern day Sellafield site in Cumbria. In the early 1950s both graphite reactor piles 1 & 2 on the Windscale site were completed and started creating plutonium for the weapons program. The fuel for these reactors was constructed at the Springfields site in Preston. (World Nuclear Association (WNA), 2013)

APPENDICES B

Tracking

Tracking in Virtual Reality is a closed loop system that feeds real world location data on the user back to the computer that can then interpret into virtual environment location data. This feedback loop allows the user to look around and even navigate the the virtual environment in the same way they would in the real world. Tracking allows for the position and orientation of real world objects to accurately transcribed to the virtual environment. The majority of tracking systems work by correlating the position of the real world object with a reference known real world location. The reference location is then set an absolute position in the virtual world. In this way the tracking systems then transfers all location and orientation data on the tracked object in the real world to the virtual world as a set of coordinates from the reference object. The closer the relationship between movement in the real world and the virtual environment are the greater the sense of Presence for the user is (SLATER et al., 1995).

There are a number of different methodologies used in tracking, often tracking systems will use a combination of methods to improve the accuracy, latency and/or stability of the system.

B.1 Time of Flight

Time of flight systems work by recording the time it takes for a signal to be received at on location after it has been sent from a separate locations. If the speed of the signal is known then the distance the signal must have travelled is simply a function of time. By using multiple receivers then the location of an object can be calculated using the triangulation of the three distances. The technique depends upon the speed of signal being constant.

Ultrasonic
A typical system using using ultrasonic signals would consist of three or more emitters on the object that is to be tracked, and another three or more receivers at the reference point. The relative positions of the receivers will be a known entity, as will the relative positions to each other of the emitters. An emitter will be a transducer, which could come in a variety of types but all of which will emit pressure waves of frequencies between 20kHz & 200MHz.

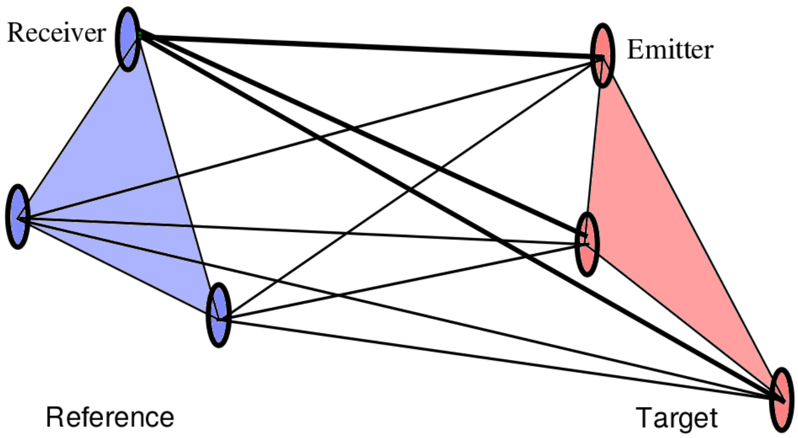


Figure B.1: Diagram of sensors and receivers showing the ToF calculation

The system works by the emitters sending sequential pulses, the pulses are picked up by the receivers. The time difference between each receiver getting the signal will give the location of the emitter as a plane, the location of the emitter on this plane is then calculated using triangulation.

This technique allows tracking of an object to 3 degrees of freedom, but get a full six degrees of freedom, that is to get the orientation of the object as well as the position, then a set of emitters is required. Each emitter will have to either fire separately from each other or use a different frequency, else the signals will interfere. Having the emitters fire in sequence will dramatically reduce the update rate, and so the resolution of the movement whereas having different frequencies will either require different sets of receivers for each frequency, which will increase cost, or decode the combined frequencies either by hardware or software. The second option, is more complicated, and arguably less expensive depending on the number of sensors required. It does come with additional latency cost, as the calculations to get the positions and orientation are more complicated.

Advantages to ultrasonic sensors and time of flight is that the emitters are inexpensive, small and lightweight. Disadvantage of this system include interference from other devices that may be emitting ultrasonics, such as hard disks and CRT monitors. The system also requires line of sight between the emitters and receivers, which can be partially mitigated by the use of more emitters spread around the tracked object.

GPS

Global Positioning Satellites (GPS)also use ToF and triangulation to calculate the position of an object. The GPS network is made of 24 geostationary satellites and 12 ground stations spread across the planet (KAPLAN et al., 2005). Each satellite has an atomic clock on board, but despite the accuracy of such a clock, corrections are still made every 30 seconds to account for the relativistic drift caused by the higher angular velocity of the satellite.
To calculate the location on earth to within 10m takes four satellites to perform ToF calculations on

a signal, this can be reduced to 1m by a method called differential GPS which uses ground stations as well as satellites (Noe et al., 1994).

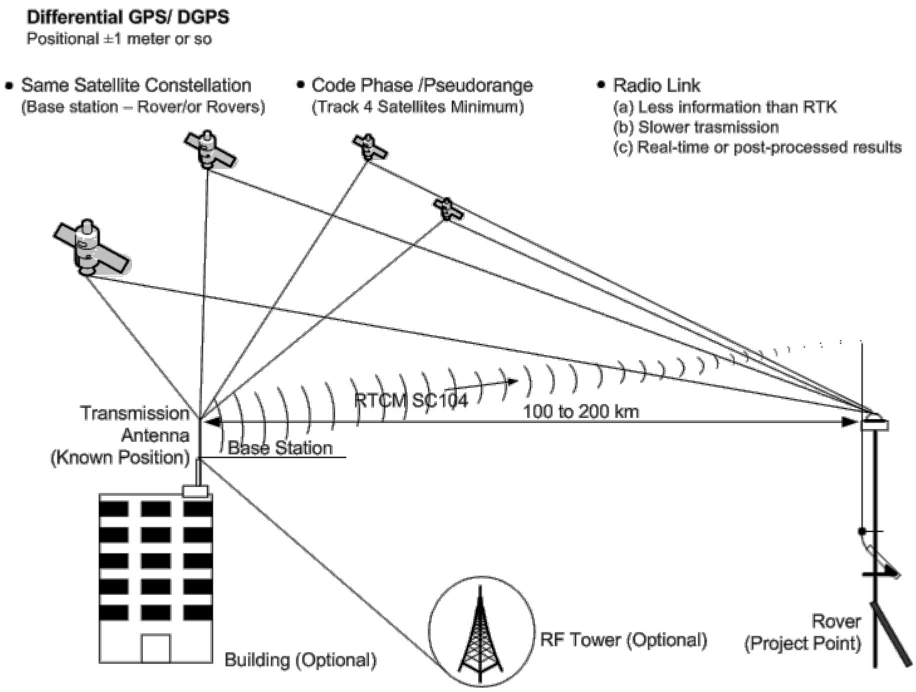


Figure B.2: Differential positioning (Van Sickle, 2001)

The disadvantages of this system is that is suffers from disruption from a variety of electro magnetic sources including, ones that take place in the atmosphere. It also gives low positional resolution.

Optical gyroscopes

Optical gyroscopes also use time of flight calculations. They come in two major types, the Fibre Optic Gyroscope (FOG)and the Ring Laser Gyroscope (RLG). The two types of gyroscope work on the same principle as mechanical gyroscopes, with a sensor for each axis. The FOG uses a simple interferometric technique to split a beam of light into two, and send the two beams round the circumference of a loop or square. When the device is not in motion the path lengths of the two beams will be the same and a simple fringe pattern will form. When the device undergoes rotation in the plane of the loop or square then one path will be longer than the other, and this will cause a different fringe pattern. The number of fringes created by this method is proportional to the angular velocity (Meyer-Arendt, 1989).

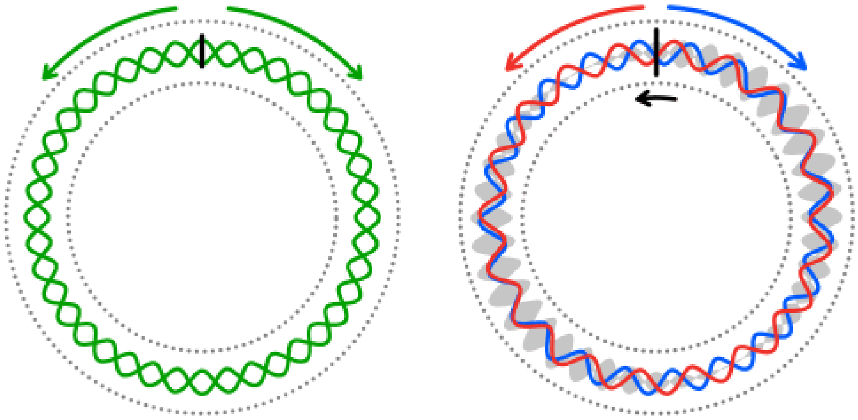


Figure B.3: Two standing wave beams experience frequency change under rotation, changing the 'beat' frequency (TEUNISSEN, 2010)

The Ring Laser Gyroscope works in a slightly different manner, in that it used two different frequencies of laser, sent in opposite directions around the same loop. Under rotation again the path lengths differ between the two beams, and the so the 'beat' frequency changes. Like the number of fringes in the FOG, the change in beat frequency is proportional to the angular velocity (FRIEDEN, 2001). This is called the Sagnac effect.

Optical gyroscopes have the advantage that they do not need to be calibrated for zero angular velocity, they also won't suffer from drifting reference axis or gimble lock which you find in mechanical gyroscopes. The gyroscope is an on board sensor and so doesn't require line of sight to a base station, however they do need to transmit the data to a base station, which depending on the method may require line of sight. A disadvantage to a gyroscope is that on their own they can only measure the rotation of an object, not its position.

B.2 Spatial scan

Spatial scanners use optical sensors like Charged Coupled Devices (CCDs) or lateral effect photo diodes. The system works by using an array of light sensors to detect the midpoint of emission of light. The systems are frequency specific and use ToF and triangulation in a similar fashion to ultrasonic devices. The method can be used in conjunction with specified markers that clearly define certain points or geometry. These markers will be reflective at the correct frequencies and be designed to reduce noise from other frequencies, which will enhance the spatial resolution the detectors are capable of. Infra red frequencies can also be used to track devices that give off heat, like animals, or vehicles. The main advantage of these systems over the similar ultrasonic devices is that they use electro-magnetic radiation that propagates at a greater and more consistent speed. There are two main variations of spatial scans, that is Outside-In and Inside-Out, which relate to where your receiver is in relation to the tracked object.

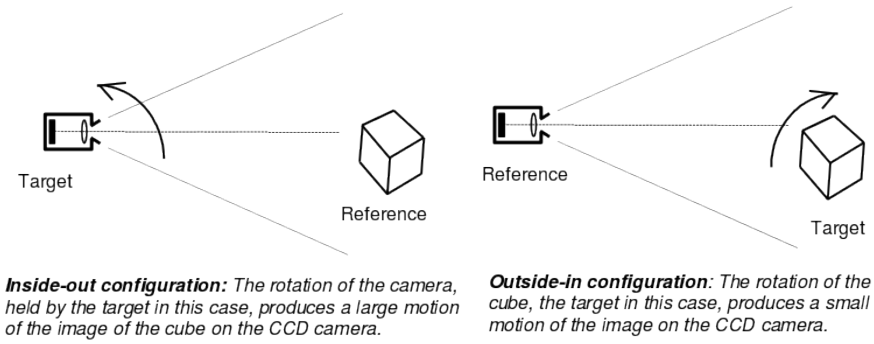


Figure B.4: Diagram of the Outside-In and Inside-Out methods of spatial scanning (ROLLAND et al., 2001)

The outside in technique provides a greater level of precision in its tracking, as it creates a greater angle of movement, so a larger spatial resolution than its counterpart.

There are also two main techniques which can be used for either variant. They are multi-scopy which is similar to ultrasonic tracking, as it uses two or more sensor arrays to track the object, and then relies on triangulation and ToF to give the location and orientation. The second technique is pattern recognition which uses a single sensor array. The methodology requires that the objects or markers geometry is known, so that any recorded 2D image of the object or marker can be identified as a distorted version of the original. This distortion will be a function of the position and orientation of the object or marker. If a minimum of three cameras are used in multi-scopy then both methods can be used to ascertain the position and orientation of the object, that is give 6 Degrees of Freedom DoF. A variation on the pattern recognition technique uses interference patterns which are projected into a known space. The interference pattern is known by the software system, so that any aberrations in fringe patterns can be attributed to spatial displacement. This technique is less common in tracking and more widely used for surface measurement.

B.3 Inertial sensing

Mechanical gyroscope
A mechanical gyroscope uses the conservation of angular momentum of a wheel to create an axis fixed for reference, around which the outer parts of the device can rotate. Three external loops connect to the motor by 'low friction' rotational encoders or potentiometers. The large mass of the motor internal creates an angular momentum significantly larger than that caused by the rotation of the supporting metal rings. Therefore encoders or potentiometers are rotating around a fixed point, so that the change in orientation of a device with a gyroscope attached can be ascertained. Simpler versions of mechanical gyroscopes use less supporting rings to create a more robust device, these devices only offer 2 DoF, so need to be used in pairs, where the reference axes are carefully aligned orthogonally. Two ring gyroscopes are more commonly used in tracking devices, as the lack of a third ring means that gimbal lock can no longer occur.

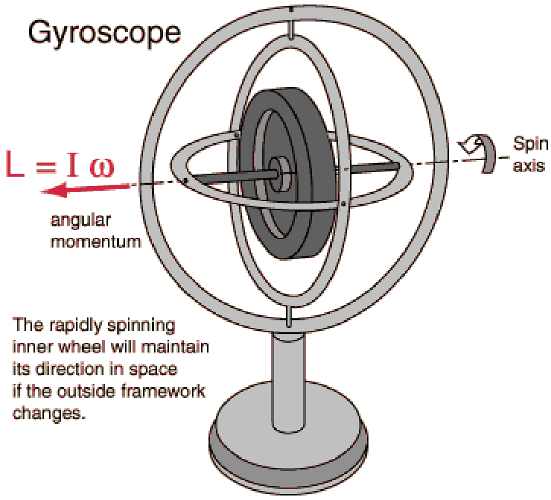


Figure B.5: Angular momentum creates a fixed axis for use as a reference (NAVE, 2012)

Using gyroscopes have the advantage that all they don't require an external point of reference, this removes the restriction of a working volume that many other tracking devices have. A gyroscope only returns information on orientation, and this can be subject to error as a truly frictionless joint is impossible. There fore with every rotation that motors pitch is affected. This can lead to an accumulation of errors, especially if the measurements are always from the reference axis. The errors can be reduced but not eliminated, by taking frequent measurements relative to the previous result rather than the original reference axis.

Accelerometer
Similar to a mechanical gyroscope an accelerometer uses mass in an inertia based system. Within an accelerometer is a mass which is attached to either piezoelectric electric crystal or another sensor that would respond under distortion. When the device undergoes acceleration, the inertia of the mass must be overcome, so a force is applied between the piezoelectric crystal and mass. The piezoelectric crystal will create a voltage which is proportional to the force applied. This measured force can be converted back to an acceleration by Newtons second law of motion. The distance moved to create the acceleration can be obtained by a double integration with respect to time. Provided an original reference point has been created, then the new location can be known. Used on their own an accelerometer can only provide positional data.

B.4 Mechanical linkages

Mechanical devices, such as arms use, potentiometers and/or encoders on joints to track the rotation at each point. If the distance between the joints is known then a location can be determined. Increasing the number of joints on a device, increases the DoF correspondingly.

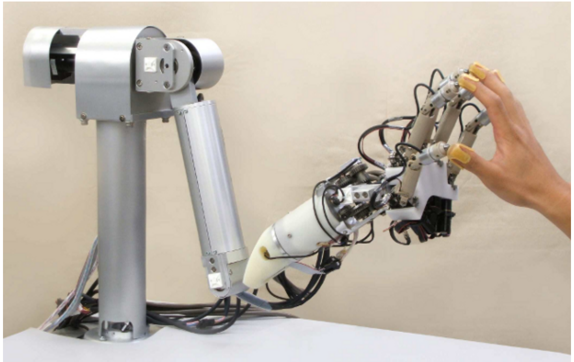


Figure B.6: Five-finger haptic interface: HIRO II, uses mechanical linkages in a multipoint haptic device (KAWASAKI et al., 2005)

Devices such as this have a limited working volume, and can restrictive for the user. They are currently one of the most common devices used for haptic feedback, as motors can be introduced at the joints to restrict or force movement back into the user.

Mechanical linkage devices need not be as complicated as the HIRO II, a simple computer mouse works as a mechanical linkage using a the recorded movement of a ball over a surface the users hand movement across a plane is tracked. Newer mice use laser optics and ToF to track the two dimensional plane, however the effect for the user is the same.

Locomotion
A subset of mechanical linkage tracking devices are locomotion devices. These devices are used to track a users movement
Phase-difference sensing
Phase difference trackers use a similar set-up to ultrasonic devices, with three emitters and three receivers. Instead of ToF the device measures the difference in phase between the emitted signal and one kept in the receiver. The technique was pioneered by Ivan Sutherland in 1968 (SUTHERLAND, 1968) for the positional tracking on his prototype HMD. A significant advantage of the phase difference technique is the ability to have a faster sample rate. This is due to the fact the signal isn't reliant on a high precision timing device. The faster sample rate also allows for a high data rate. It is possible to measure the phase difference from all emitters at once. This gives the method an improved spatial resolution and higher accuracy to the ToF methods. The disadvantage of interference at ultra sonic frequencies still exists. There is also a maximum speed that an object can be tracked at. Any movement greater than the measured wavelength of the signal performed between measurements will induce errors in the measurement. In this respect the faster an object needs to be tracked, then the higher the sample rate required.

B.5 Direct field sensing

Direct field sensing relies upon the induced effect of flux caused by placing a ferromagnetic coil into a magnetic (B) field. At the reference point is a base station that emits a strong B field, or in practise three orthogonal B fields. A receiver or receivers are then moved through this field, this creates a measurable magnetic flux within their own three orthogonal sensors. The magnetic flux is a function of both the distances and orientation of the base station emitters and receivers. The receivers are light weight and compact as they do not need to emit strong EM fields themselves, but instead serve a passive function.

The base station uses with AC or DC current to create a large enough B field for tracked volume. Using AC create a sinusoidal variation in the created B field, which will cause the creation of a current in the receivers. This is an advantage as measuring the current in a device is less complex than measuring the magnetic flux. However it can lead to the creation of eddy currents on surfaces of other metal objects. These eddy current will in turn have their own associated B fields which will distort the base stations, and so the validity of the tracking.

Magnetrons, allow for measurements of a constant magnetic flux to be made (MILNE et al., 1996). When a pulsed DC powered B field is created, the change in the B field created eddy currents in the surrounding metal surfaces. Once the field is established there is no more variation in the bisecting B-field of the metals and so the eddy currents subside. The longer the time between pulses the less the strength of the eddy currents and their corresponding B fields. Therefore measurements of the receivers flux is taken at the last possible moment before a second pulse. This time frame is dictated by the required sample rate or acquisition time of the system. The faster the acquisition time or sample rate the greater the disruption in the B field caused by eddy currents. If the metal objects with induced eddy currents were stationary, then their B fields could be compensated for by calibrating the receivers within the adjusted fields.

APPENDICES C

DAA Cell Guides

C.1 Work Instructions

EngD Project

A Bi-Directional data link for Virtual Environments

Author:

Stephen Reddish

s.reddish@namrc.co.uk

December 12, 2016

1 Check List

- Cable Bundles
  - Black CAT-5 cable - **10m**
  - Blue CAT-5 cable - **10m**
  - Green CAT-5 cable - **10m**
  - Grey CAT-5 cable - **10m**
  - Orange CAT-5 cable - **10m**
  - Pink CAT-5 cable - **10m**
  - Purple CAT-5 cable - **10m**
  - Red CAT-5 cable - **10m**
  - Cable bindings - **black x 16**

- Assembly One - **940mm x 365mm x 100mm**
  - Kettle Box **220mm x 220mm x 250mm**
    - Kettle Weld Bridges
    - One **75mm x 210mm x 3mm**
    - Two **65mm x 205mm x 3mm**
    - Three **85mm x 205mm x 3mm**
  - Amazon LO2016 **300mm x 230mm x 170mm**
  - Amazon DI2016 **340mm x 250mm x 150mm**
  - 2 x Box Hangars
- Assembly Two - **940mm x 365mm x 100mm**
  - SubAssembly **470mm x 240mm x 60mm**
    - Staedler **150mm x 70mm x 25mm**
    - ND HighLighters **150mm 120mm x 25mm**
    - ND BallPoints **70mm x 70mm x 150mm**
    - ND WhiteMarkers **40mm x 105mm x 140mm**
  - Amazon E7 **460mm x 250mm x 245mm**
    - Amazon E7 Weld Bridges
    - One **195mm x 95mm x 3mm**
    - Two **195mm x 100mm x 3mm**
    - Three **205mm x 85mm x 3mm**
  - 2 x Box Hangars
- Assembly Three - **940mm x 365mm x 100mm**
  - LSA Glass Box **190mm x 190mm x 125mm**
  - Amazon SPL **310mm x 400mm x 210mm**
  - 2 x Box Hangars

2 Assembly Sequence

2.1 Assembly One

The complete assembly one is shown in figure 3

1. Attach Kettle Box to Main Assembly (Figure 4)
2. Attach Kettle Weld Bridges x 3 to Kettle box (Figure 5 6 7)
3. Attach Amazon LO2016 box to Main Assembly (Figure 8)
4. Attach Amazon DI2016 box to Main Assembly (Figure 9)
5. Attach 2 x box hangars to Main Assembly box (Figure 10)

2.2 Assembly Two

The complete assembly two is shown in figure 11

1. Attach Amazon E7 box to Main Assembly (Figure 17)
2. Attach E7 Weld Bridges x3 to Amazon E7 box (Figure 18 19 20)
3. Attach Staedler box to Sub Assembly (Figure 13)
4. Attach ND HighLighter box to Sub Assembly (Figure 14)
5. Attach ND BallPoints box to Sub Assembly (Figure 16)
6. Attach ND WhiteMarkers box to Sub Assembly (Figure 15)
7. Attach Sub Assembly to Main Assembly (Figure 12)
8. Attach 2 x box hangers to Main Assembly box (Figure 21)

2.3 Assembly Three

The complete assembly two is shown in figure 22

1. Attach LSA Glass Box to Main Assembly (Figure 23)
2. Attach Amazon SPL box to Main Assembly (Figure 24)
3. Attach 2 x box hangers to Main Assembly box (Figure 25)

2.4 Cable Looms

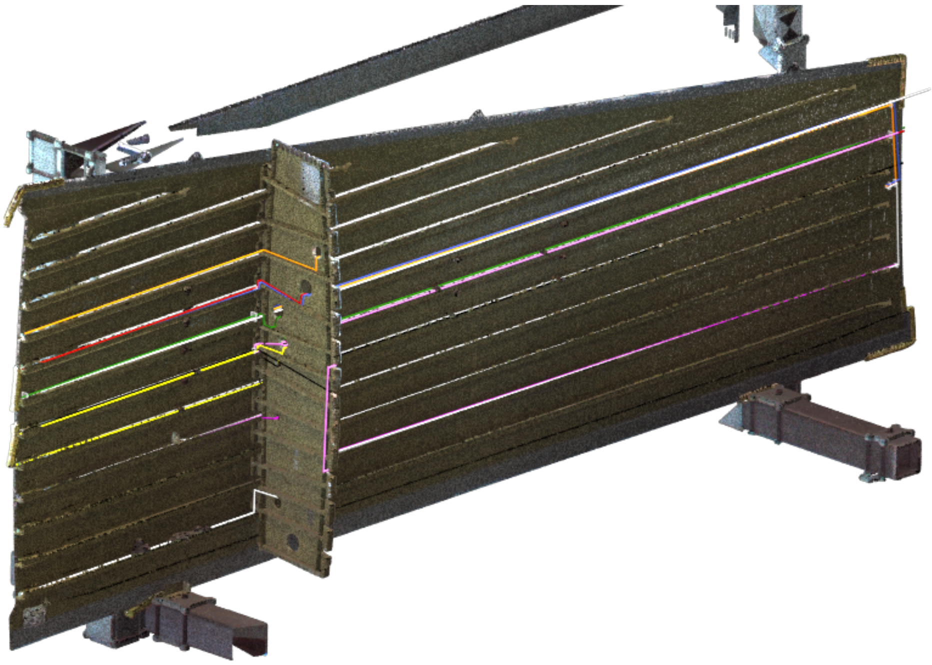


Figure 1: Rendered image of cable looms on wing skin

The wing skin is made up of 12 horizontal sections 26, and a vertical section acting as a divider. The divider is further separated into 13 sections 27.

2.4.1 Orange Cable

The orange coloured cable takes the route through the wing from left to right detailed below and in Figure 28:

1. 4th wing horizontal to stantion
2. Pass through circular hole in stantions 3rd horizontal
3. 6th wing horizontal to wing end
4. Pass through circular hole in wing end on 9th horizontal

2.4.2 Red Cable

The red coloured cable takes the route through the wing from left to right detailed below and in Figure 29:

1. 5th wing horizontal to stantion
2. Pass through circular hole in stantions 4th horizontal
3. 7th wing horizontal to wing end

2.4.3 Blue Cable

The blue coloured cable takes the route through the wing from left to right detailed below and in Figure 30:

1. 5th wing horizontal to stantion
2. Pass through circular hole in stantions 4th horizontal
3. 6th wing horizontal to wing end
4. Pass through circular hole in wing end on 9th horizontal

2.4.4 Green Cable

The green coloured cable takes the route through the wing from left to right detailed below and in Figure **??**:

1. 6th wing horizontal to stantion
2. Pass through circular hole in stantions 5th horizontal
3. 7th wing horizontal to wing end

2.4.5 White Cable

The white coloured cable takes the route through the wing from left to right detailed below and in Figure 31:

1. 6th wing horizontal to stantion
2. Pass through behind the stantion to wing end

2.4.6 Black Cable

The black coloured cable takes the route through the wing from left to right detailed below and in Figure 32:

1. 7th wing horizontal to stantion
2. Pass around stantion end on the 7th horizontal
3. Returns through circular hole in the stantions 7th horizontal
4. Pass through behind the stantion under the 7th wing horizontal
5. 8th wing horizontal to wing end

2.4.7 Yellow Cable

The yellow coloured cable takes the route through the wing from left to right detailed below and in Figure 33:

1. 7th wing horizontal to stantion
2. Pass through circular hole in the stantions 6th horizontal
3. Return through behind the stantion in the stantions 7th horizontal
4. 8th wing horizontal to left hand wing end

2.4.8 Purple Cable

The purple coloured cable takes the route through the wing from left to right detailed below and in Figure 34:

1. 9th wing horizontal to stantion
2. Pass through circular hole in the stantions 9th horizontal
3. 11th wing horizontal to left hand wing end

2.4.9 Pink Cable

The pink coloured cable takes the route through the wing from left to right detailed below and in Figure 35:

1. 9th wing horizontal to stantion
2. Pass behind the stantion at the stantions 9th horizontal
3. Return through the gap on edge of the stantions 10th horizontal

4. Pass up to the the gap on the edge of the stantions 6th horizontal
5. Return through the circular hole in the stantions 6th horizontal
6. Pass behind the stantion at the stantions 6th horizontal
7. 7th wing horizontal to left hand wing end

2.4.10 Grey Cable

The grey coloured cable takes the route through the wing from left to right detailed below and in Figure 36:

1. 12th wing horizontal to stantion
2. Pass through circular hole in the stantions 13th horizontal
3. 11th wing horizontal to left hand wing en
4. Pass through circular hole in wing end on 9th horizontal

2.4.11 Cable Binds

Cable bindings should be attached to cables at the points listed below and shown in Figure 37:

1. Left hand wing end; red and blue cables
2. Left hand wing end; green and grey cables
3. Left hand wing end; black and yellow cables
4. Left hand wing end; pink and purple cables

5. Left hand side of the wing stantion; red and blue cables
6. Left hand side of the wing stantion; green and grey cables
7. Left hand side of the wing stantion; black and yellow cables
8. Left hand side of the wing stantion; pink and purple cables

9. Right hand side of the wing stantion; blue, orange and white cables
10. Right hand side of the wing stantion; green, red and pink cables

11. Right hand side of the wing stantion; purple and grey cables
12. Right hand wing end; blue, orange and white cables
13. Right hand wing end; green, red and pink cables
14. Right hand wing end; blue, orange and grey
15. Right hand wing end hole; blue, orange and grey
16. Right hand wing end; purple and grey

2.5 Final Assembly

A reference image of the assemblies on the wing with measurements Figure 38

1. Attach Assembly One to Wing 150mm from the wing stantion
2. Attach Assembly Two to Wing 300mm from Assembly One
3. Attach Assembly Three to Wing 300mm from Assembly Two

3 Diagrams

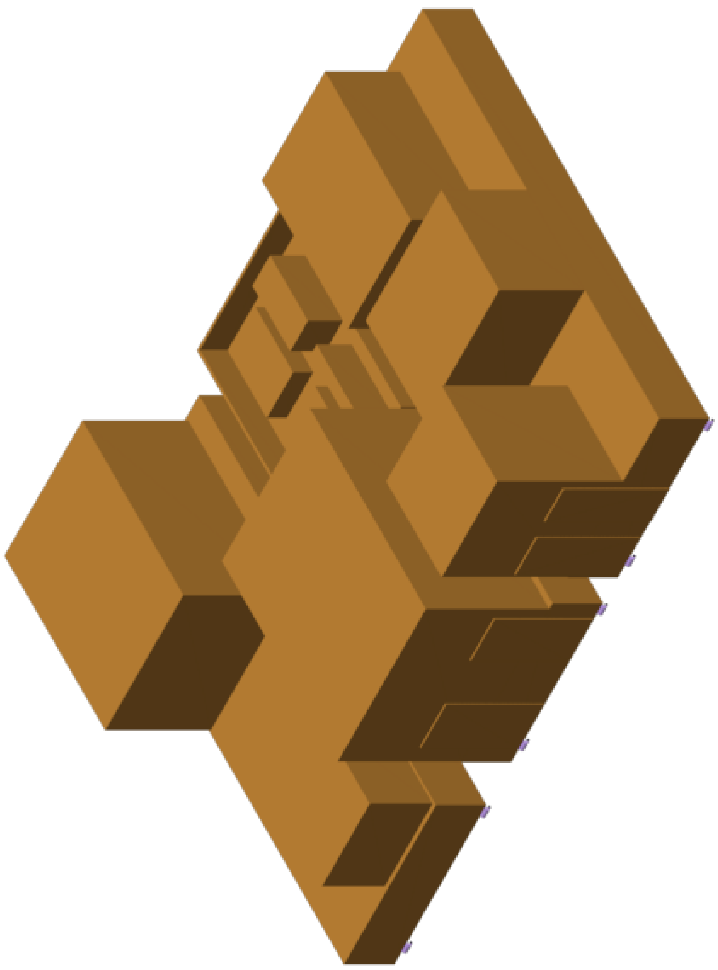


Figure 2: Rendered image of Assemblies One, Two and Three

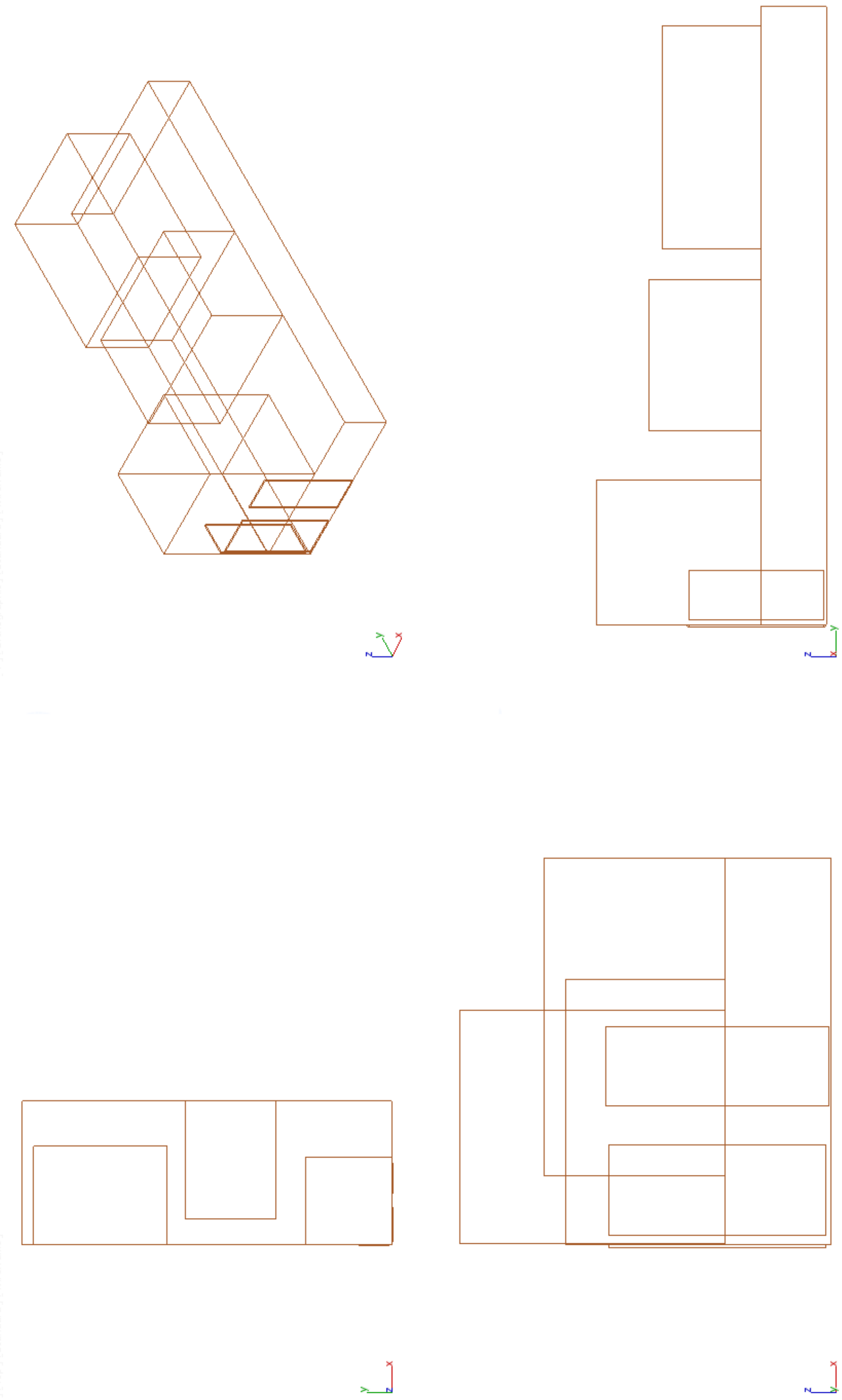


Figure 3: Orthographic render of Assembly One

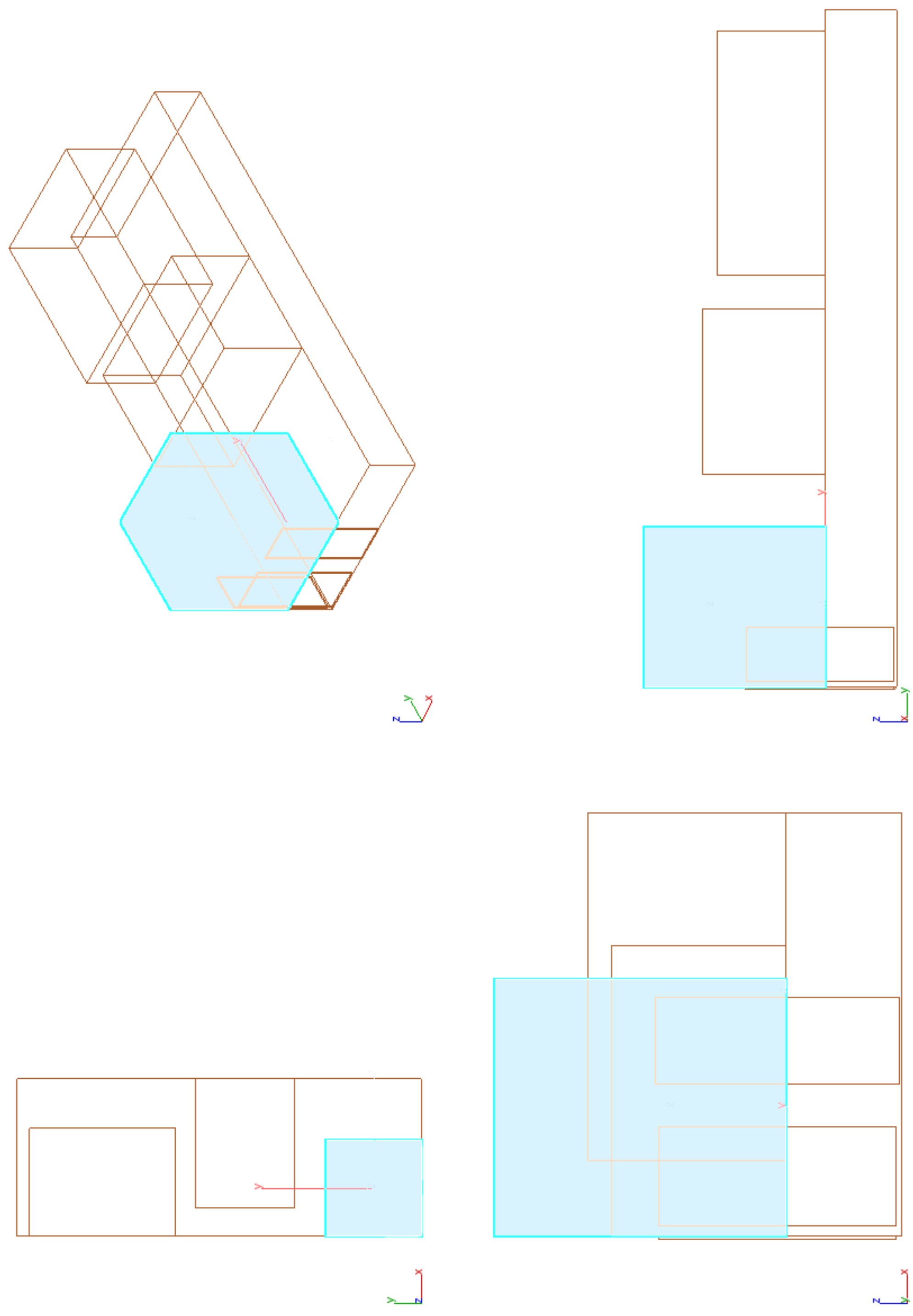


Figure 4: Orthographic render of Assembly one, Kettle Box

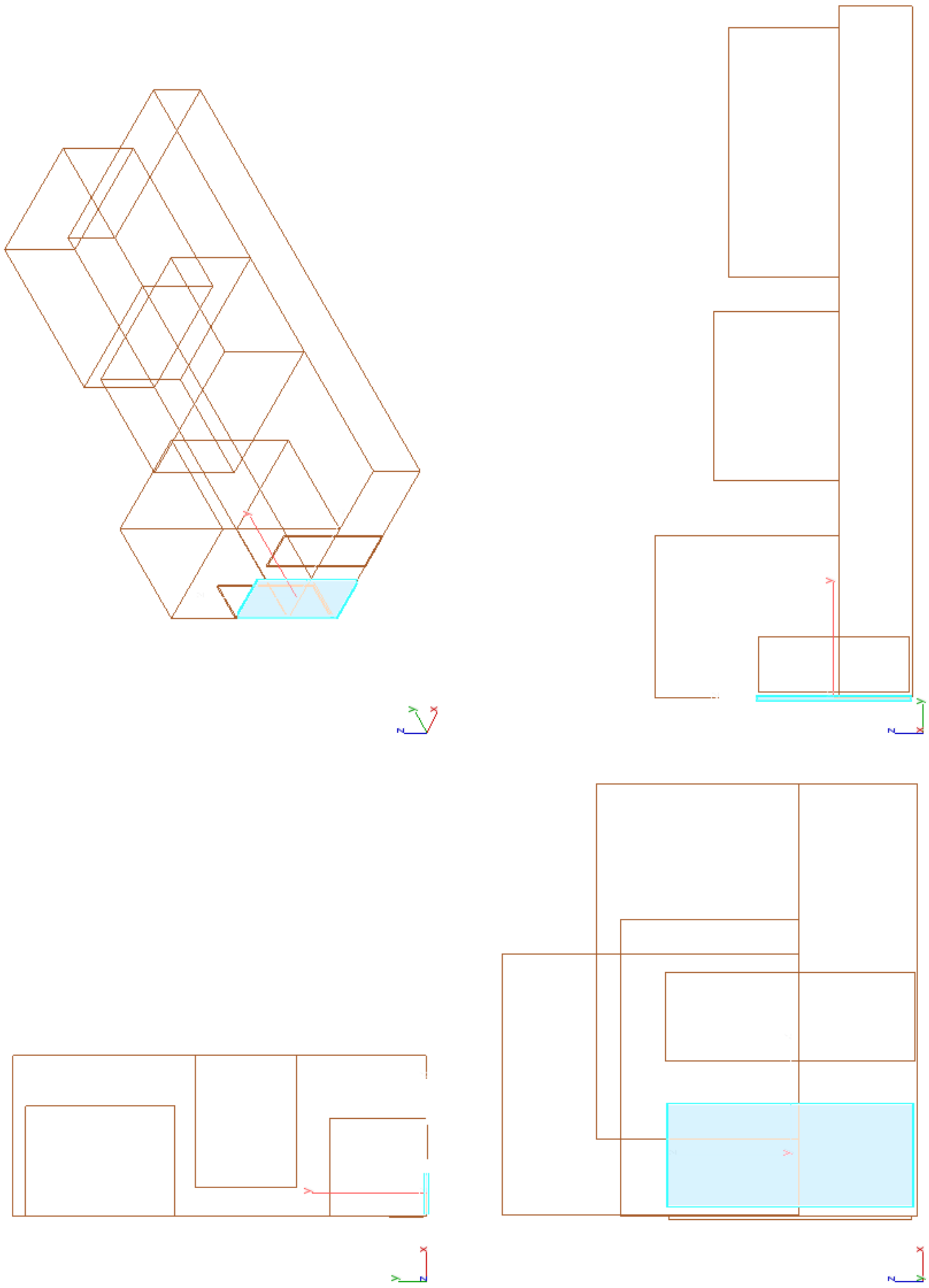


Figure 5: Orthographic render of Assembly one, Kettle Weld Bridge One

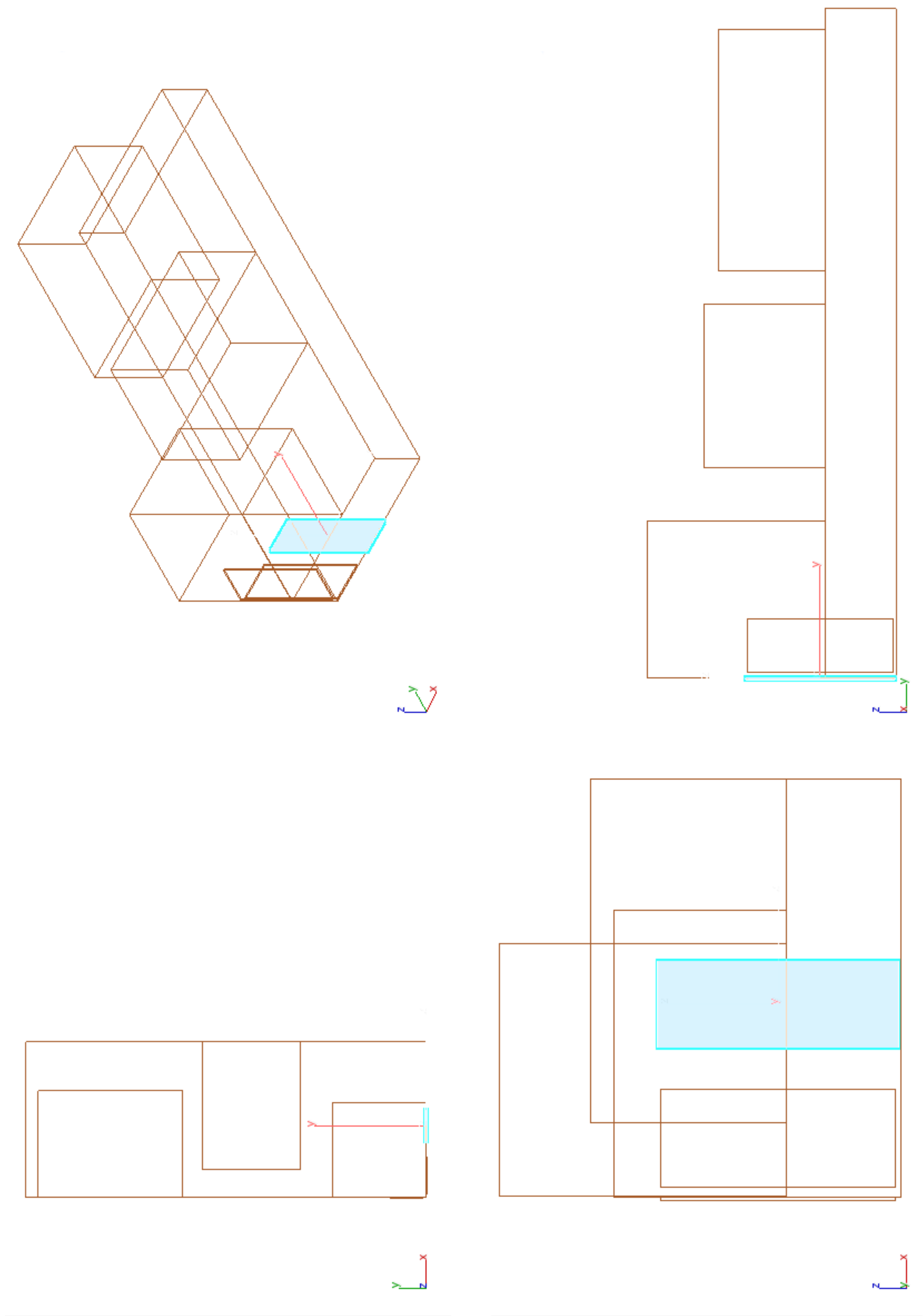


Figure 6: Orthographic render of Assembly one, Kettle Weld Bridge Two

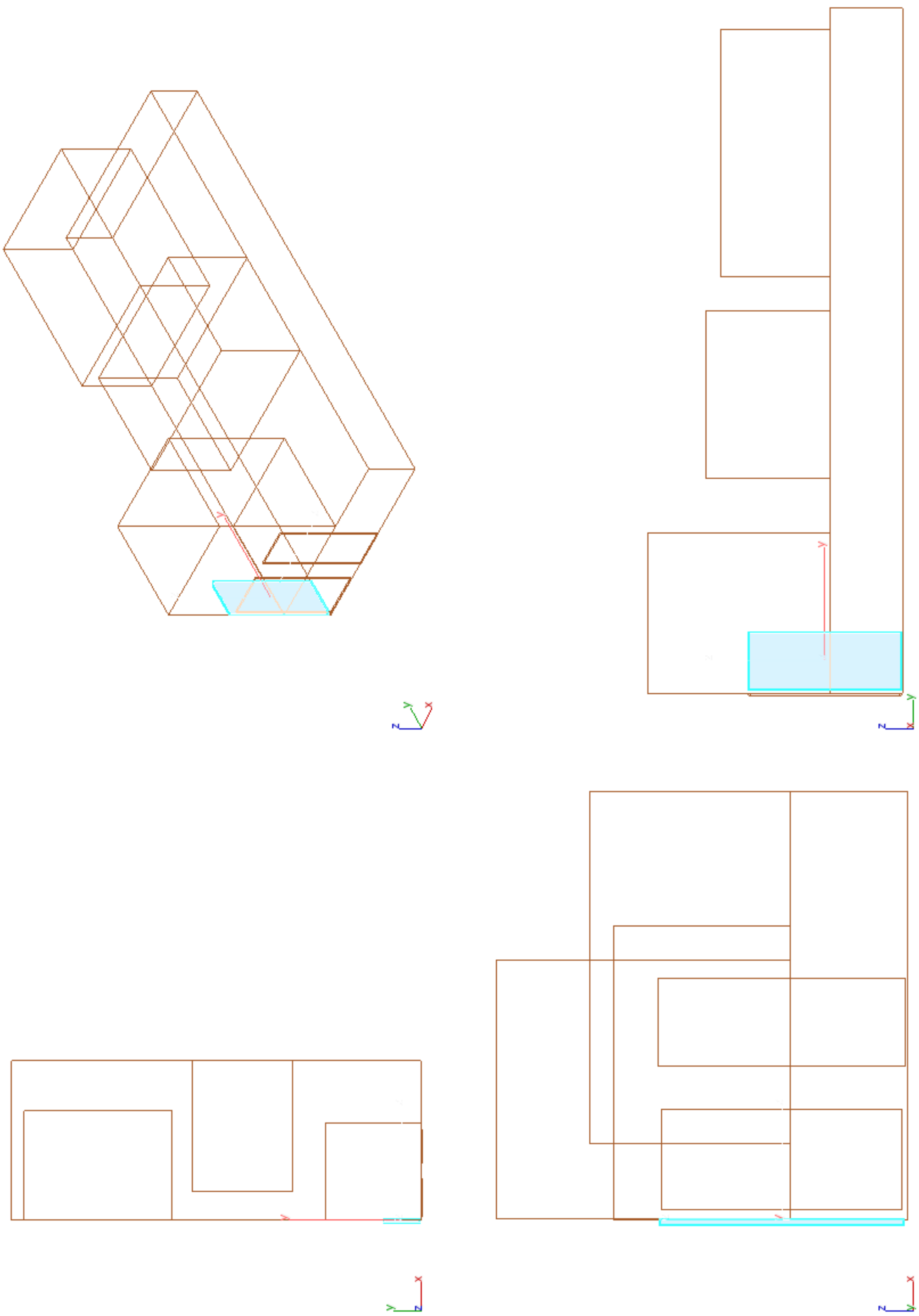


Figure 7: Orthographic render of Assembly one, Kettle Weld Bridge Three

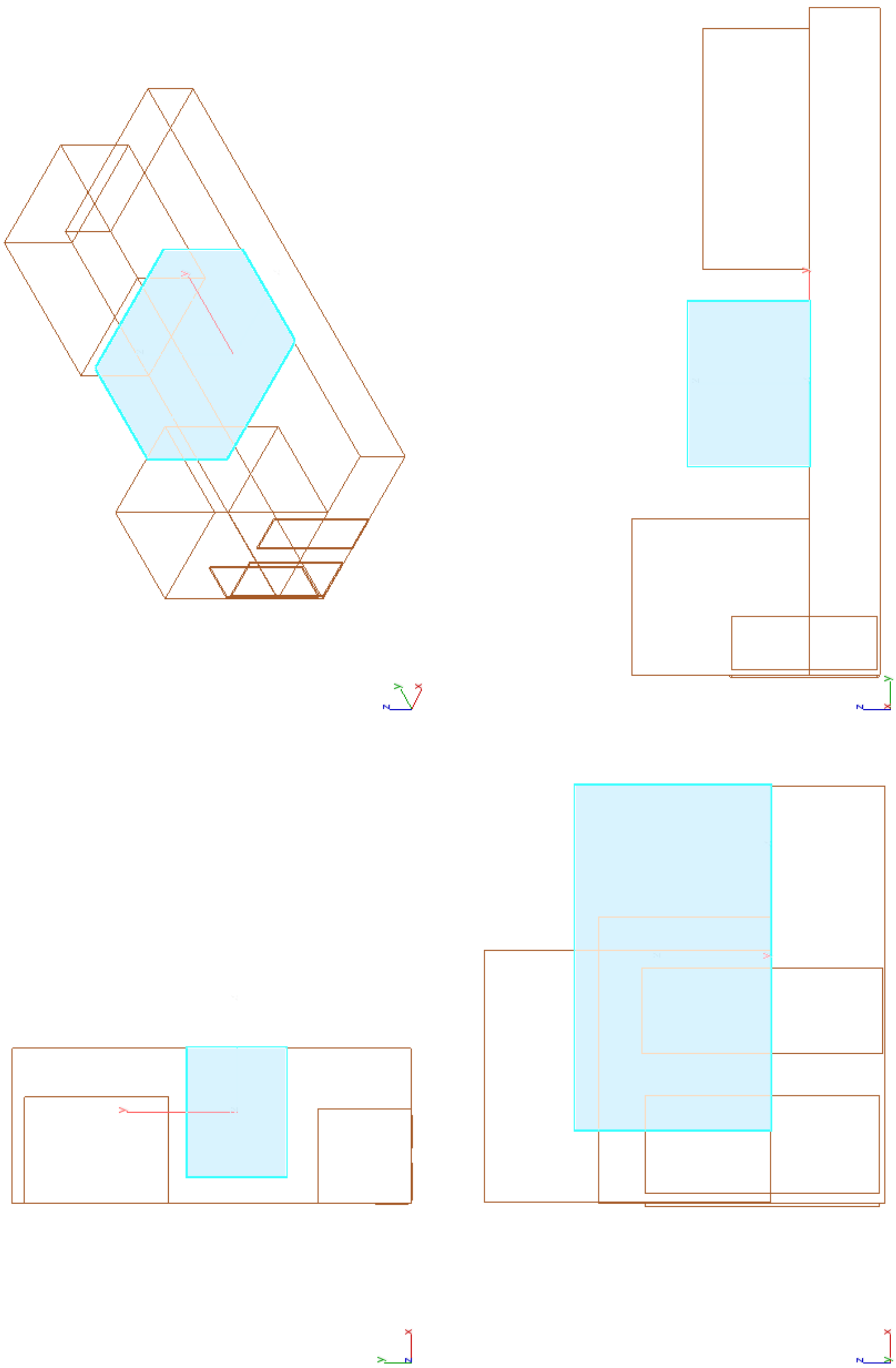


Figure 8: Orthographic render of Assembly one, Amazon Box LO2016

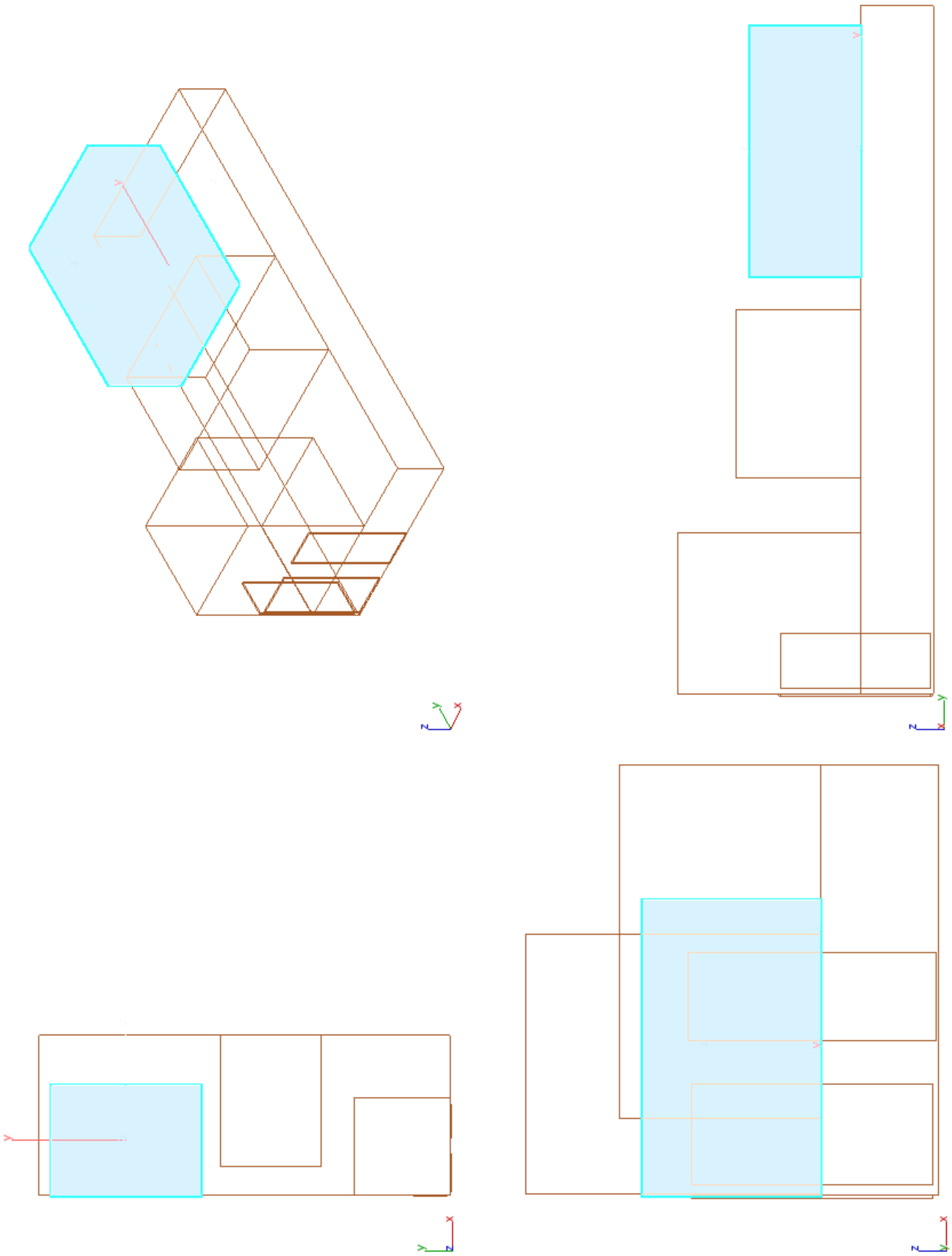


Figure 9: Orthographic render of Assembly one, Amazon Box DI2016

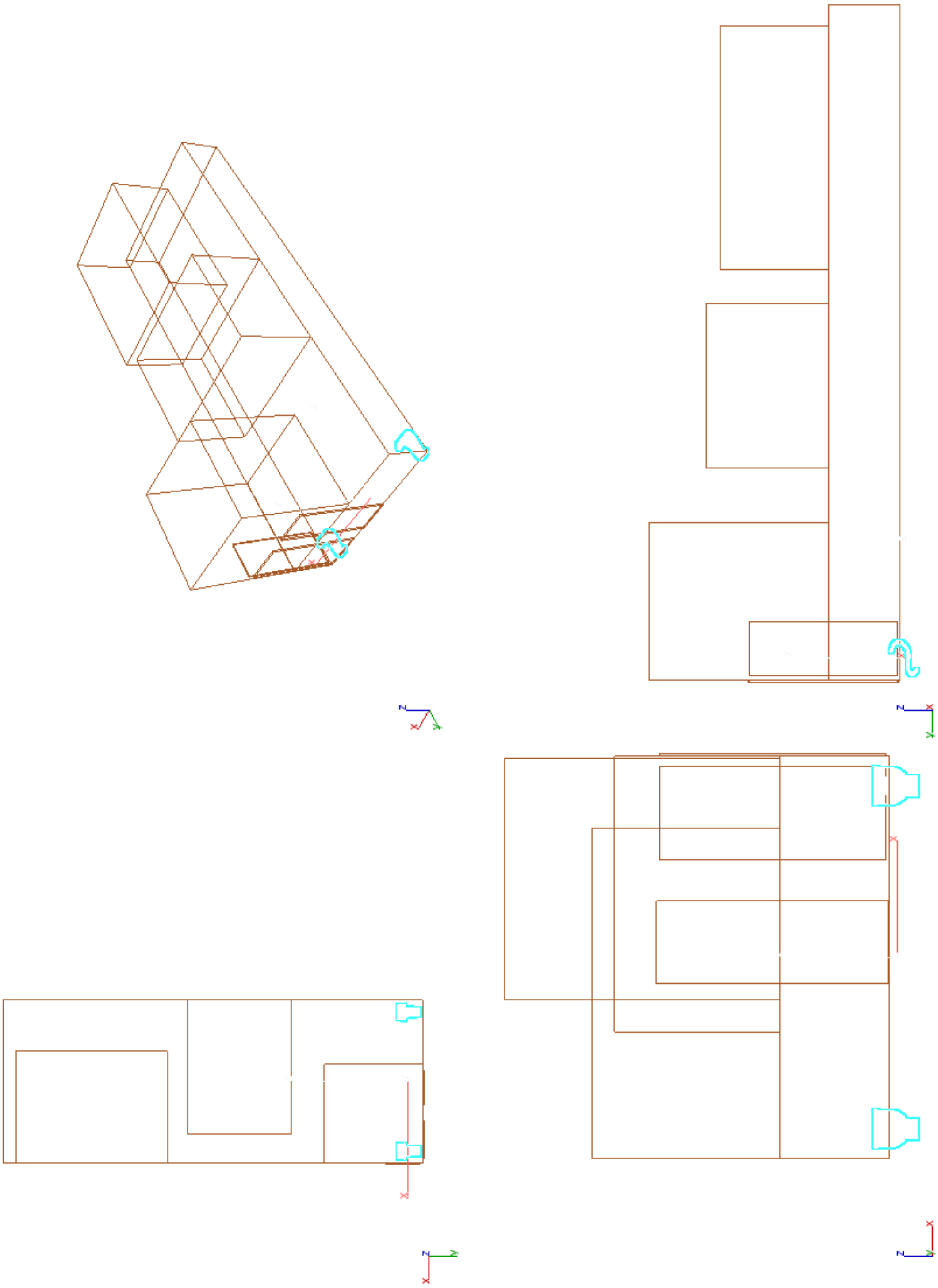


Figure 10: Orthographic render of Assembly One, Support Hooks

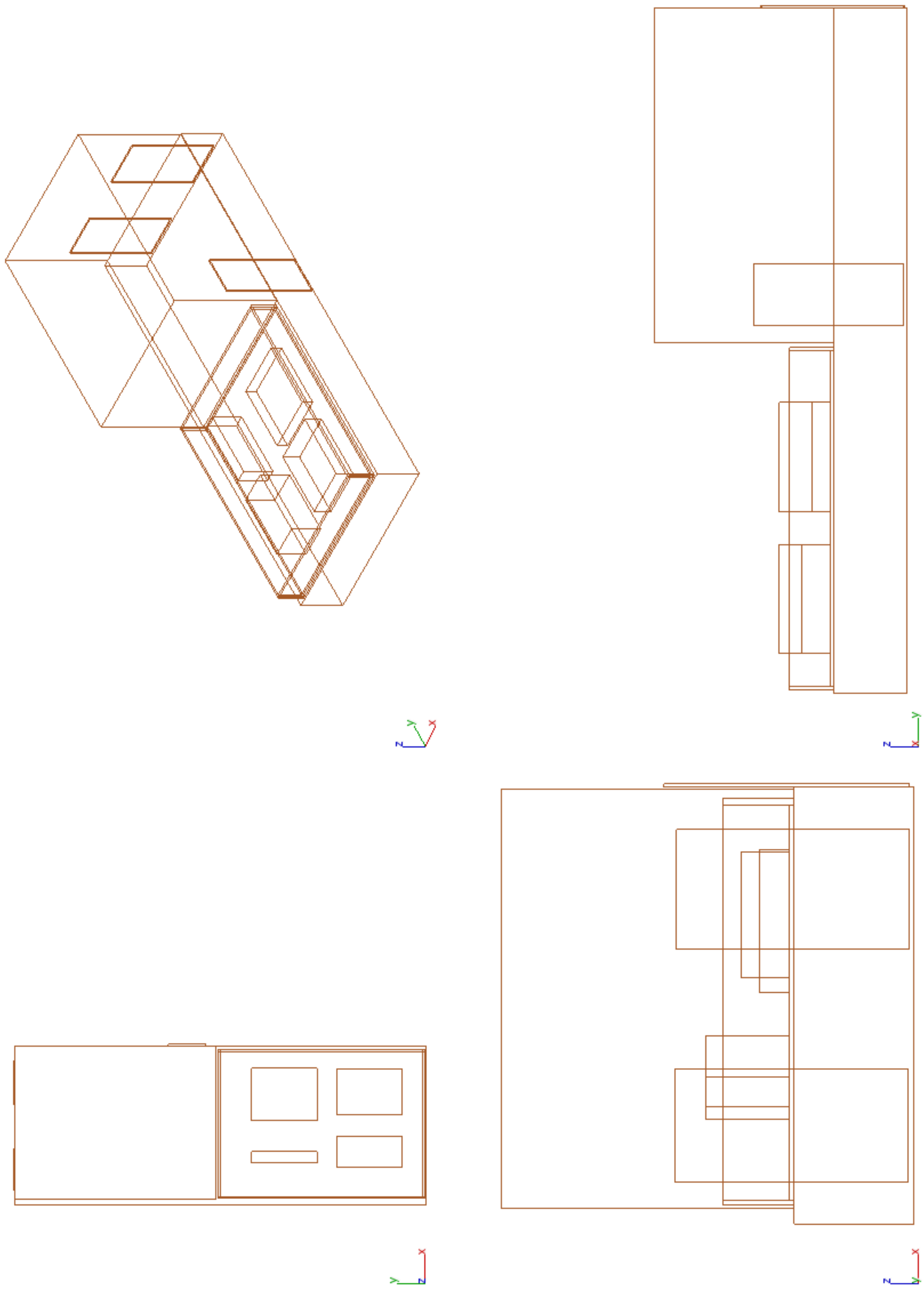


Figure 11: Orthographic render of Assembly Two

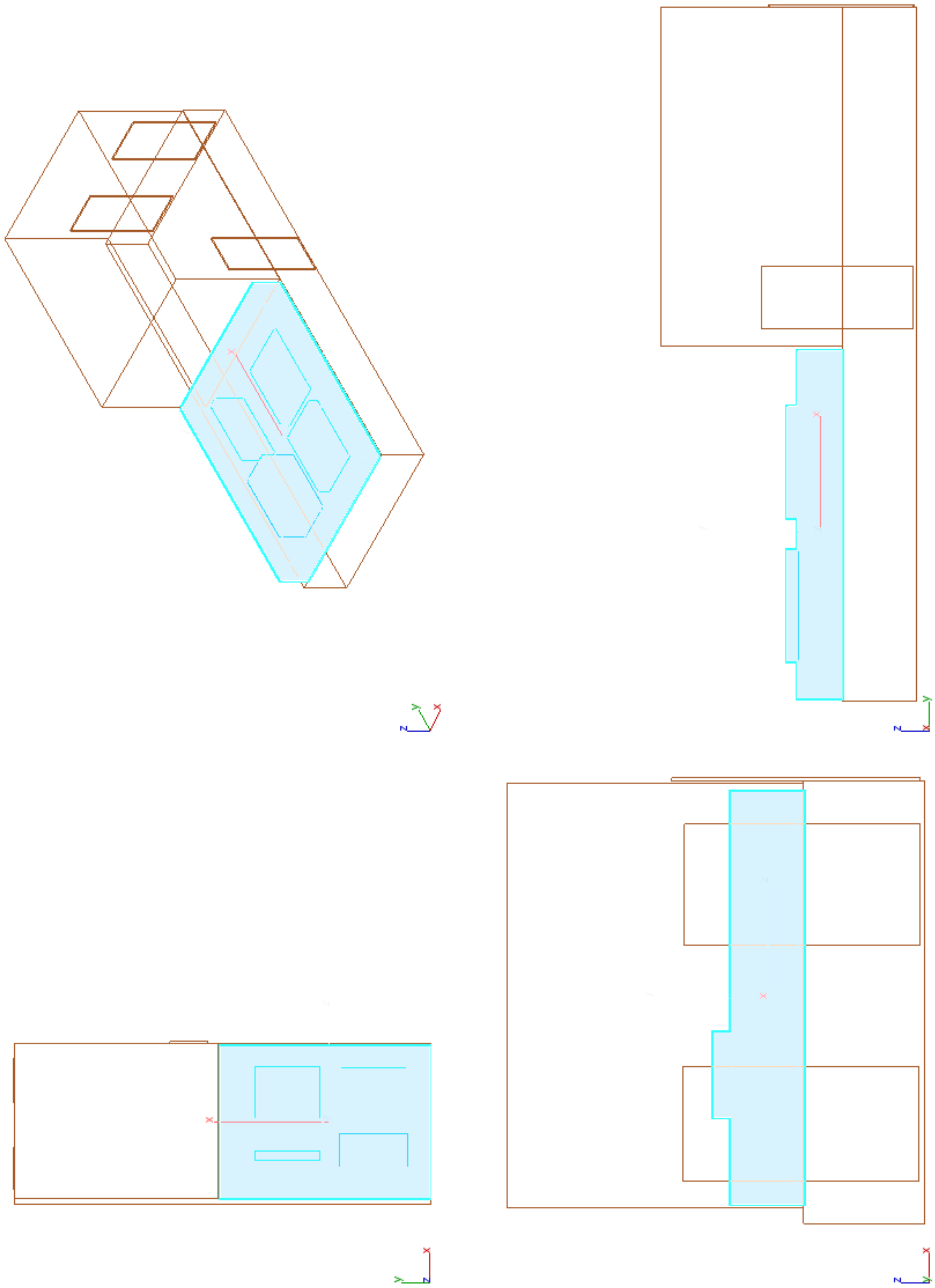


Figure 12: Orthographic render of Assembly Two, Sub-Assembly

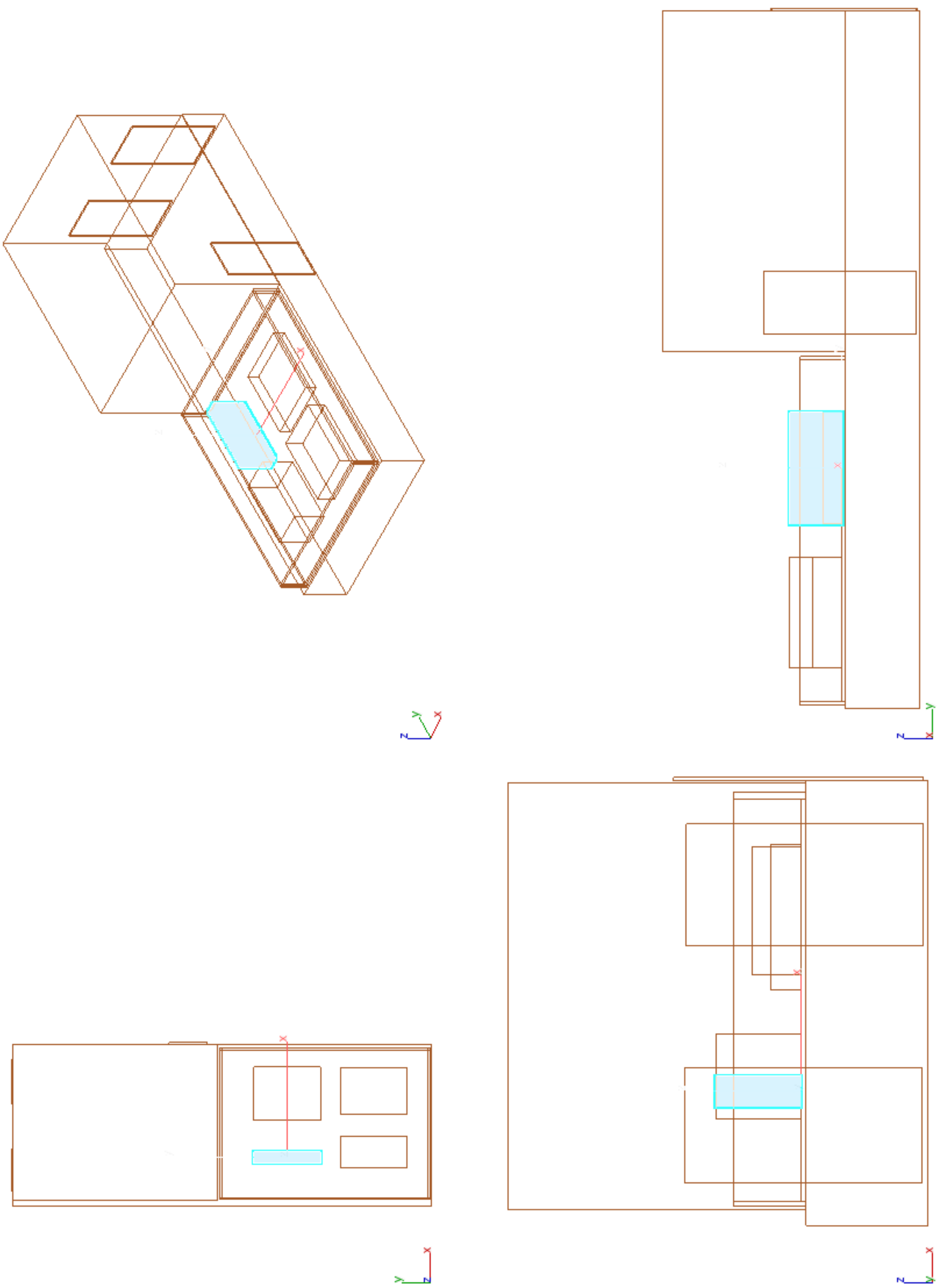


Figure 13: Orthographic render of Assembly Two, Sub-Assembly Staedler Box

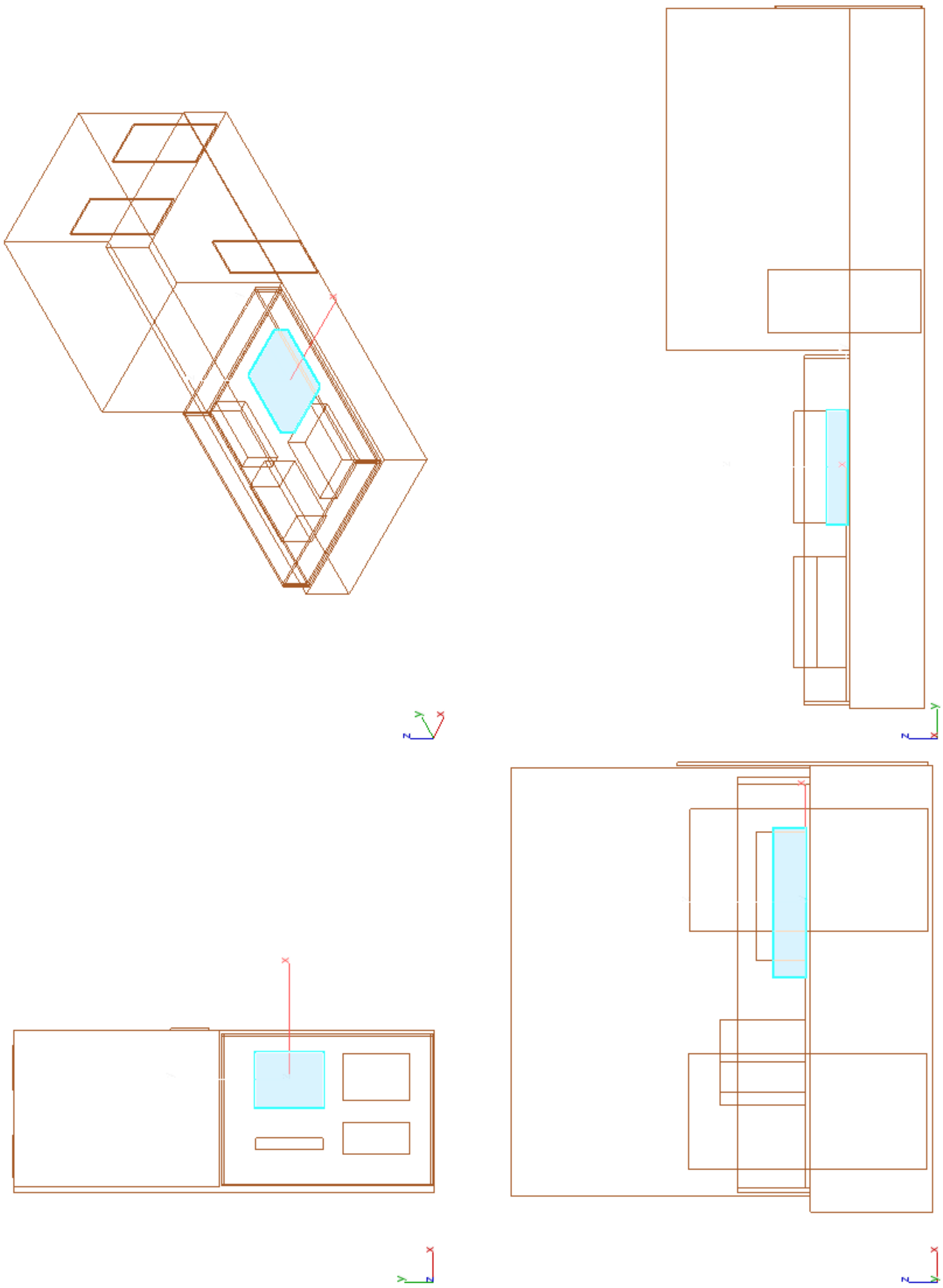


Figure 14: Orthographic render of Assembly Two, Sub-Assembly ND High Lighters box

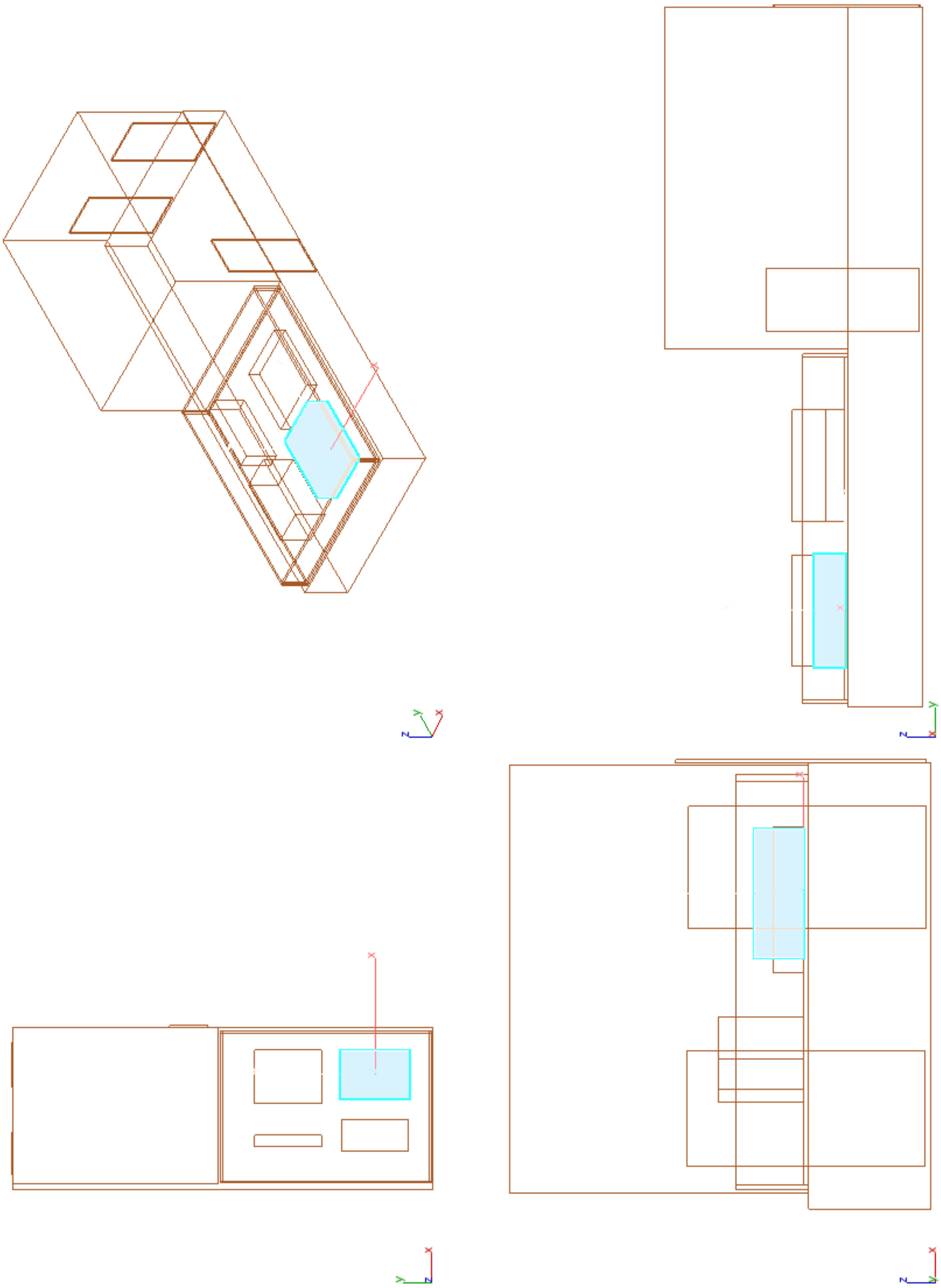


Figure 15: Orthographic render of Assembly Two, Sub-Assembly ND WhiteMarker box

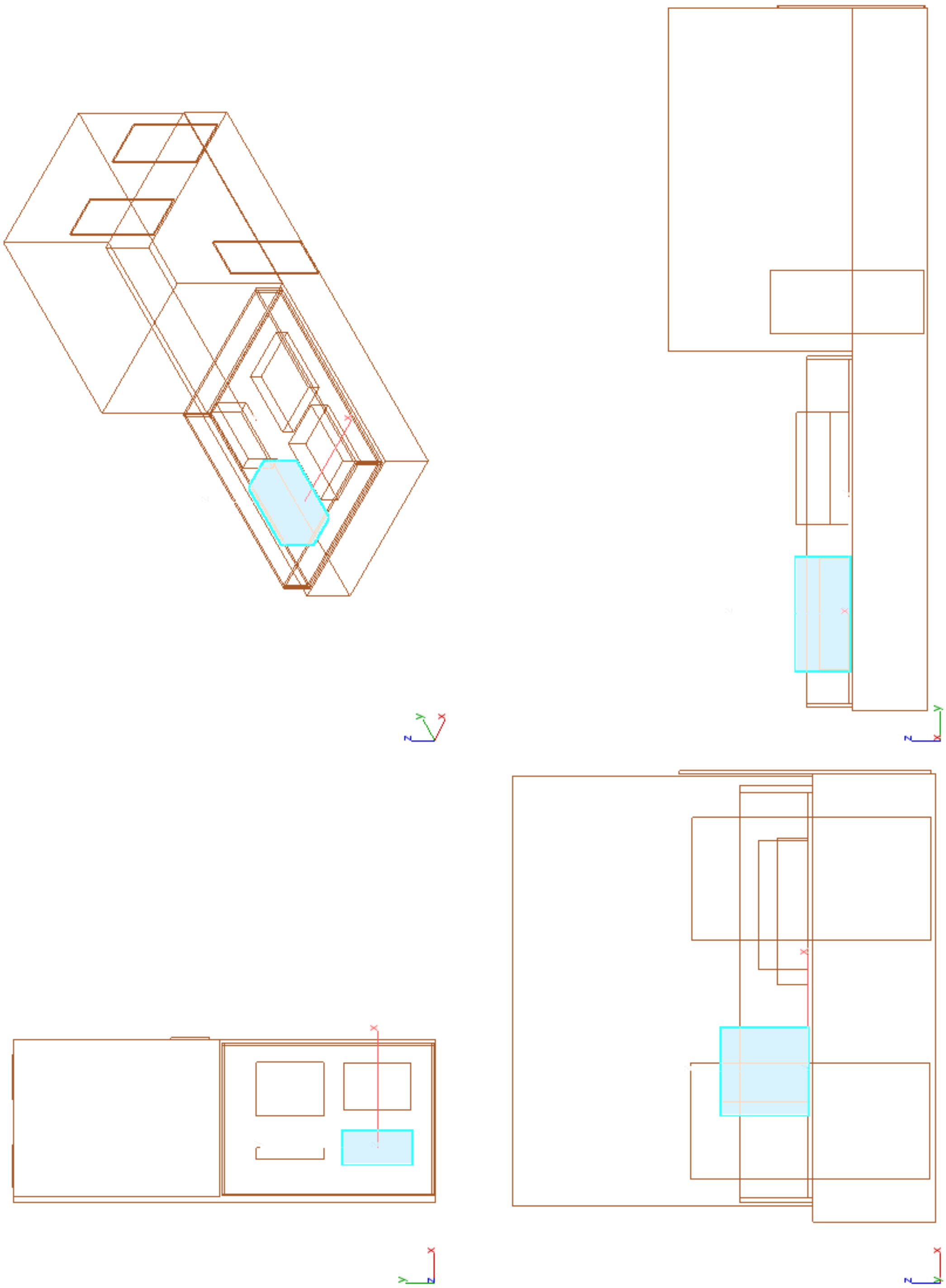


Figure 16: Orthographic render of Assembly Two, Sub-Assembly ND BallPoint box

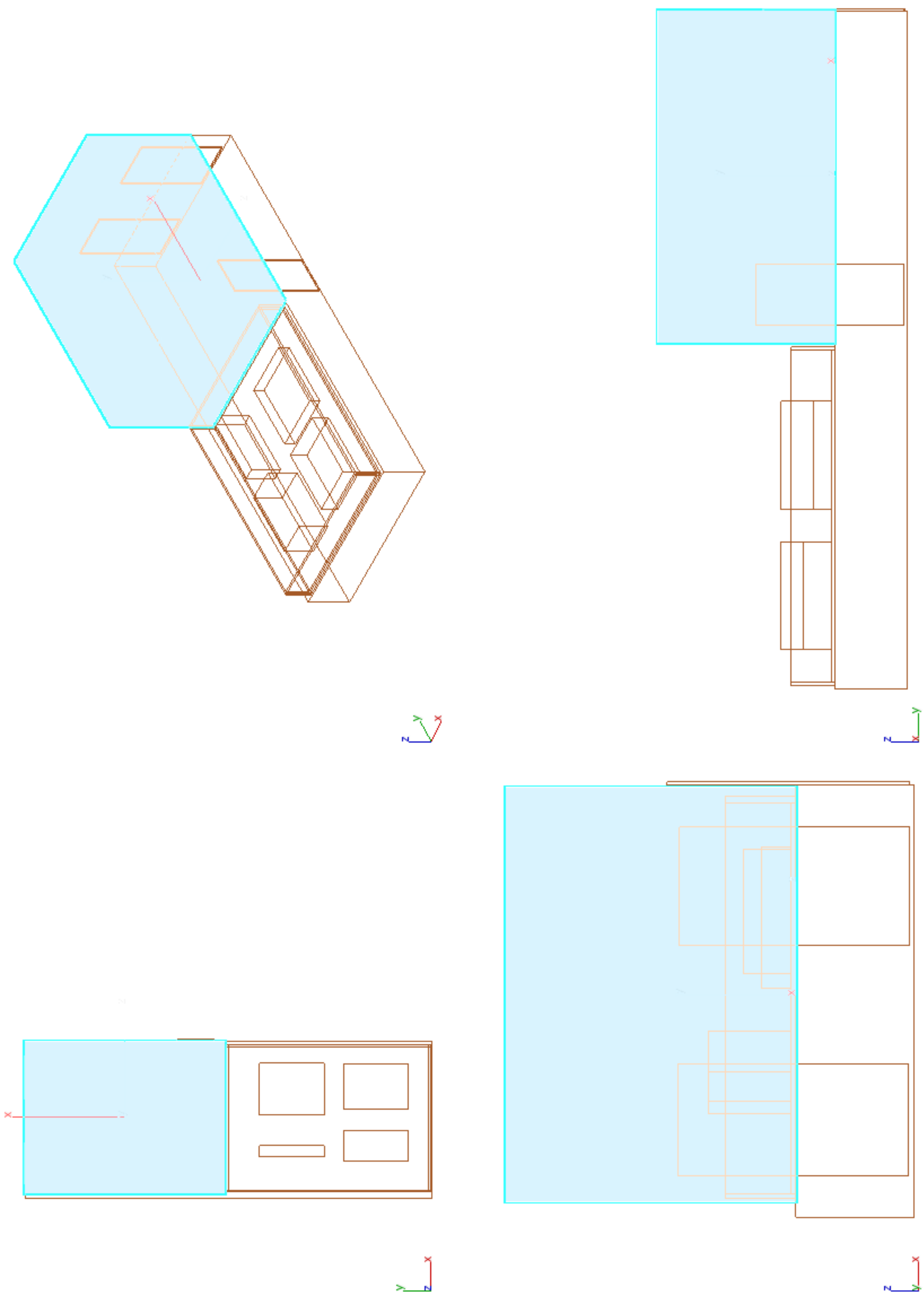


Figure 17: Orthographic render of Assembly Two, Amazon E7 box

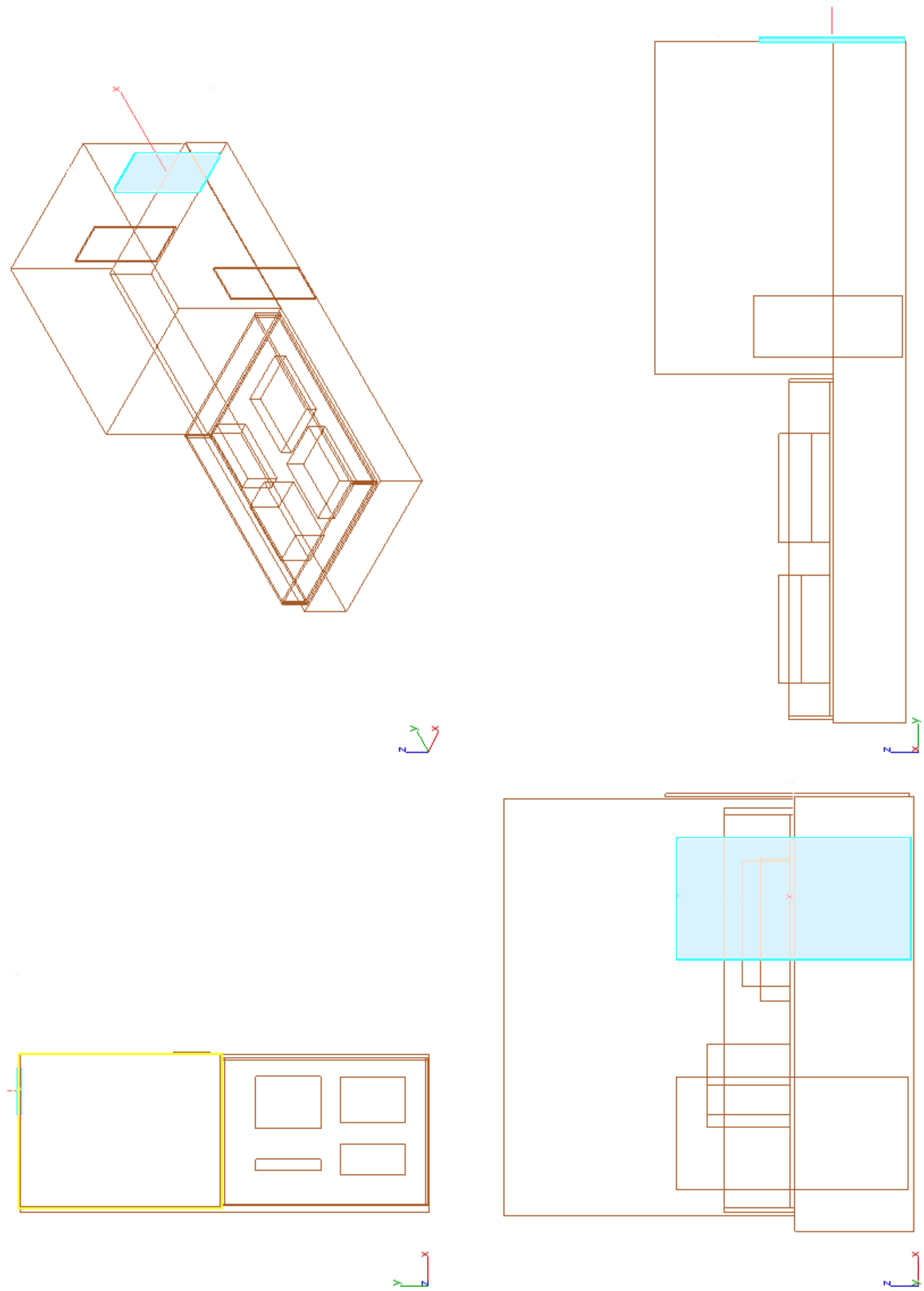


Figure 18: Orthographic render of Assembly Two, Amazon E7 box Weld Bridge One

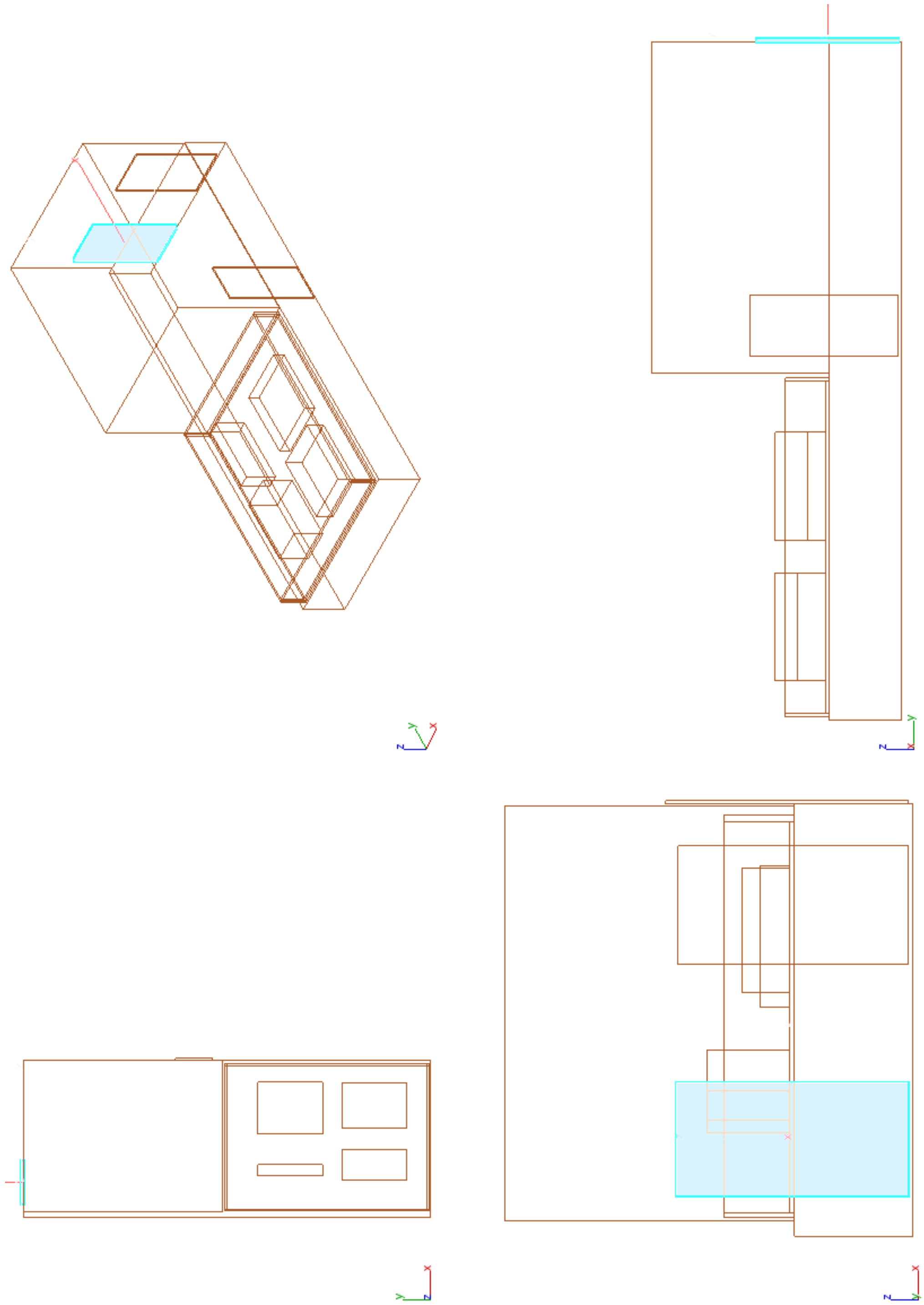


Figure 19: Orthographic render of Assembly Two, Amazon E7 box Weld Bridge Two

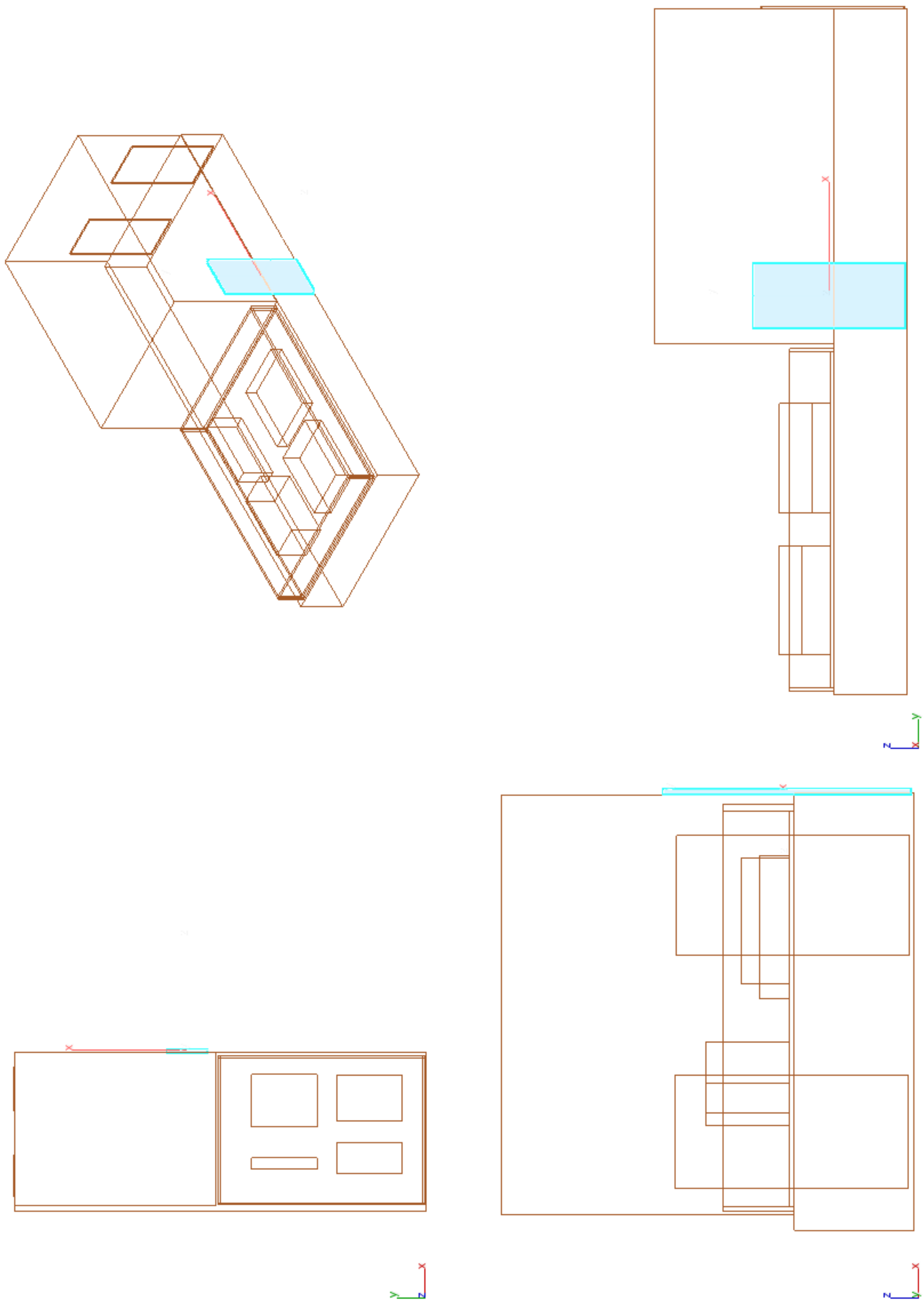


Figure 20: Orthographic render of Assembly Two, Amazon E7 box Weld Bridge Three

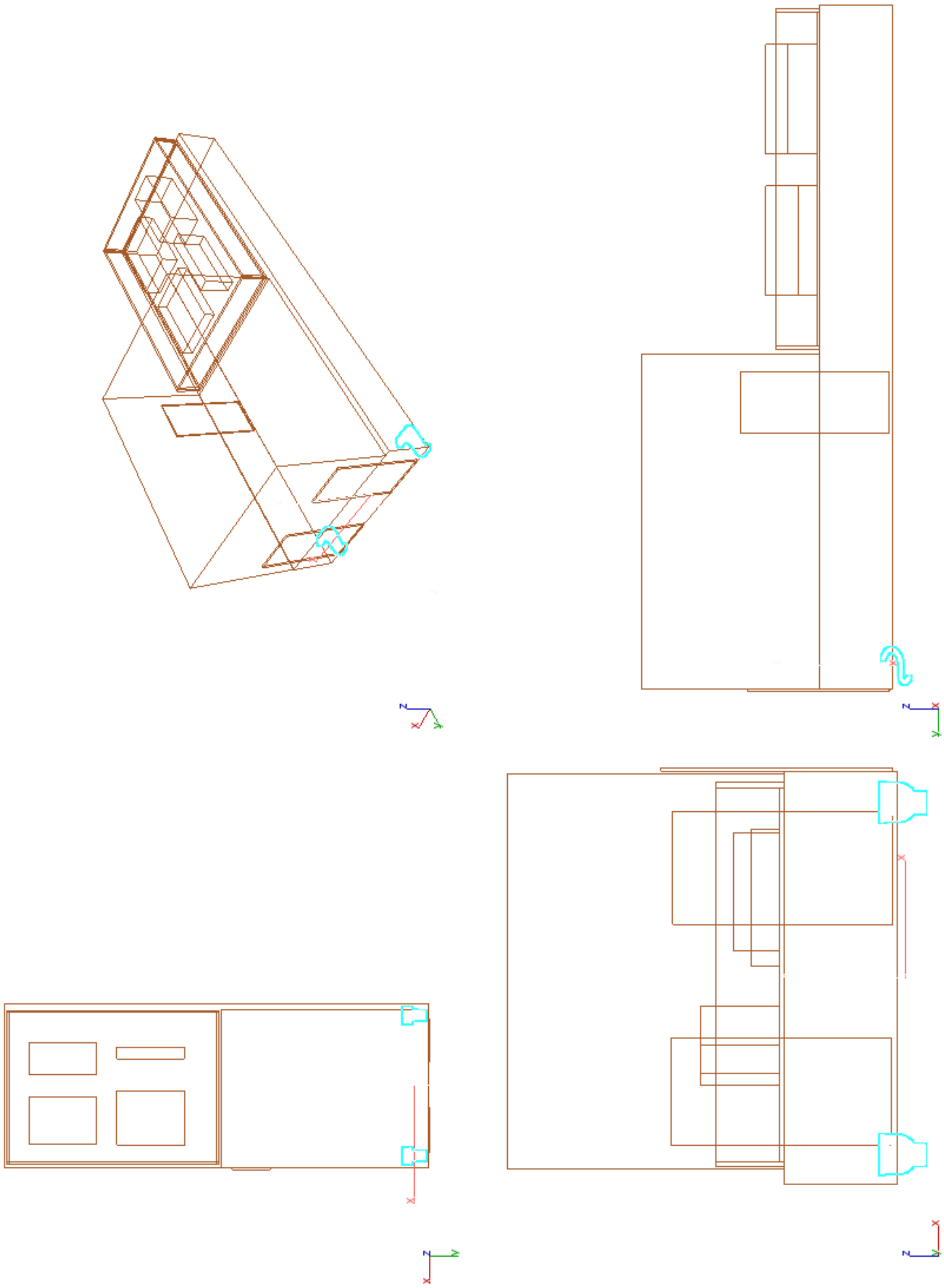


Figure 21: Orthographic render of Assembly Two, Support Hooks

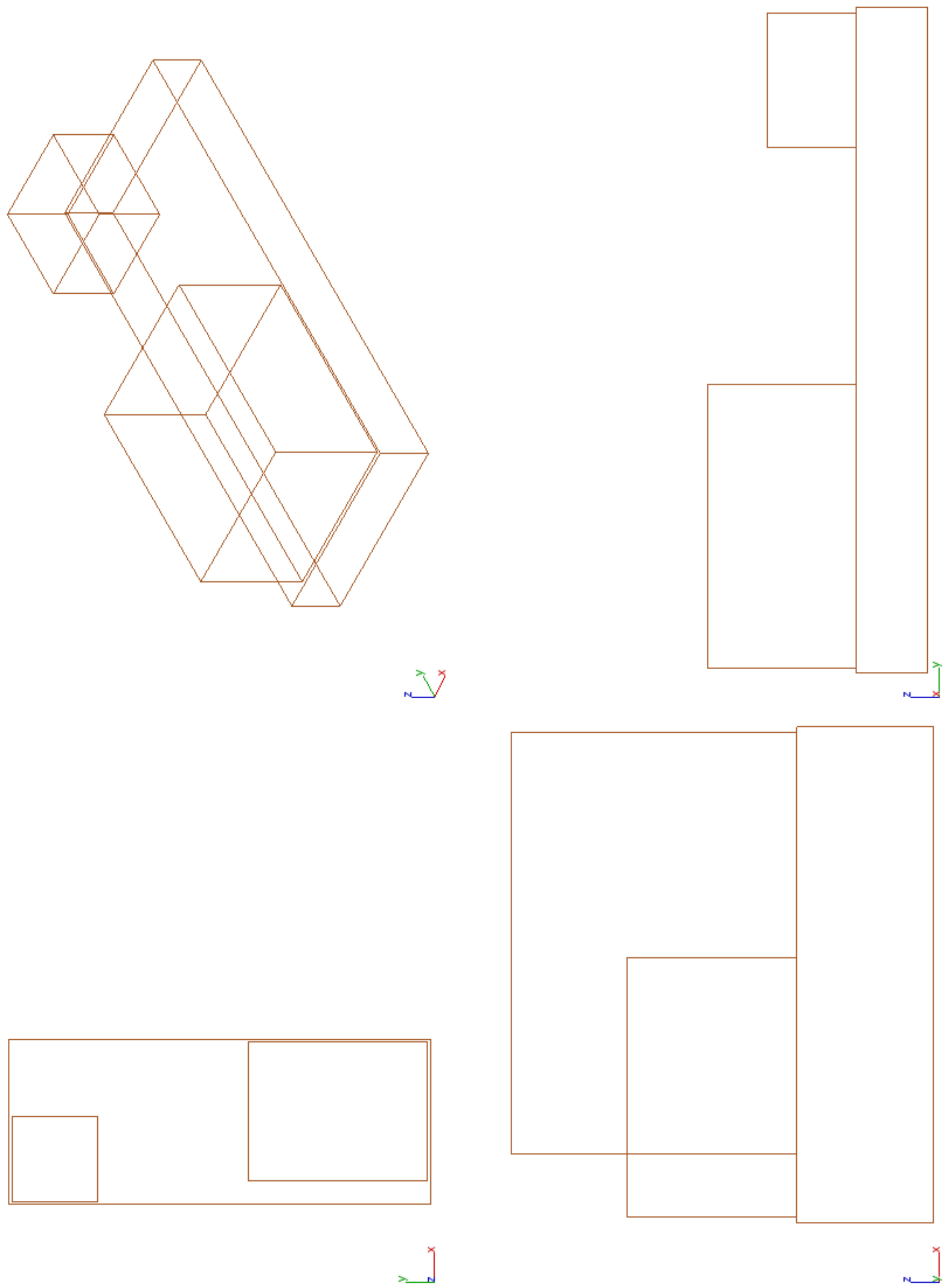


Figure 22: Orthographic render of Assembly Three

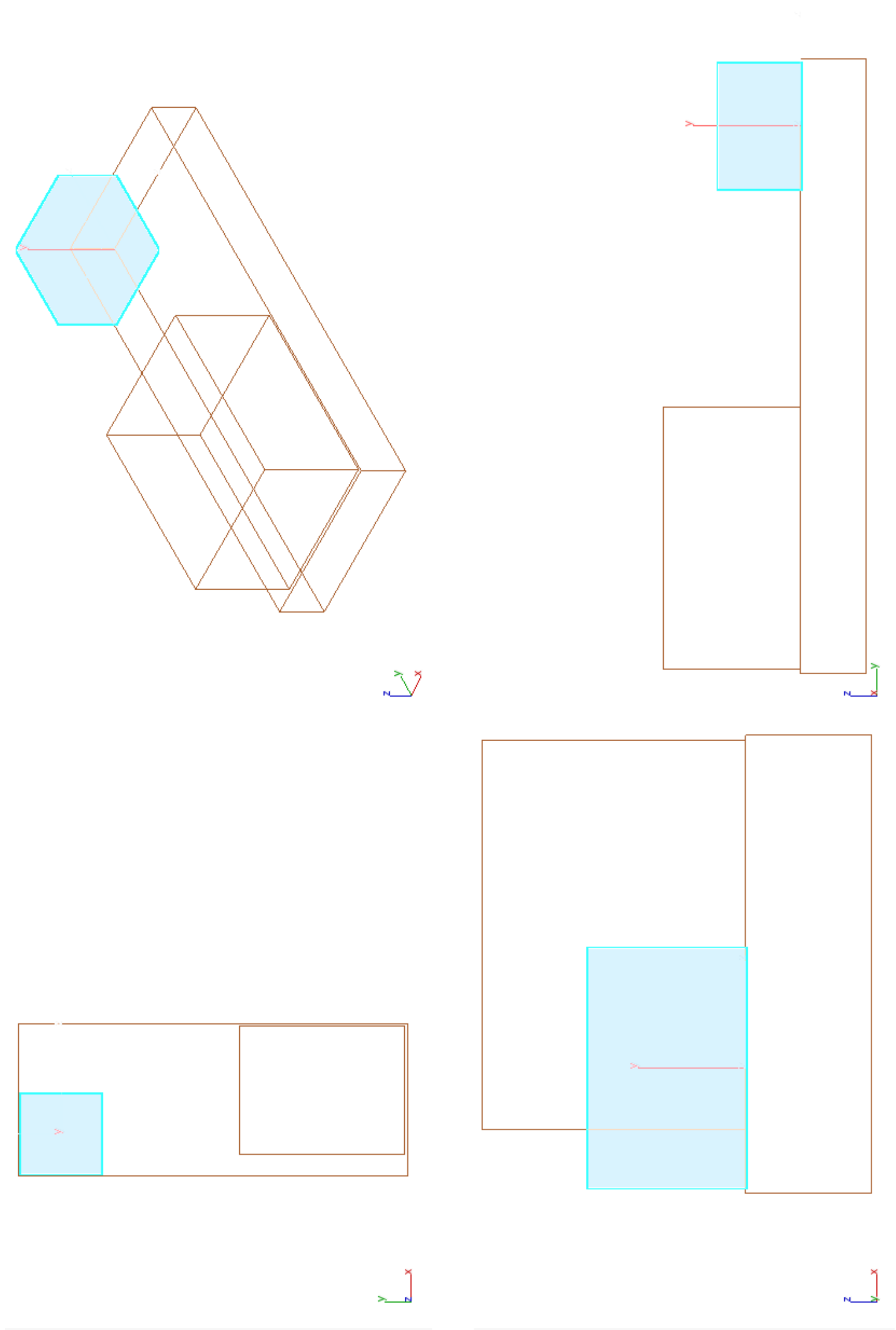


Figure 23: Orthographic render of Assembly Three, LSA Box

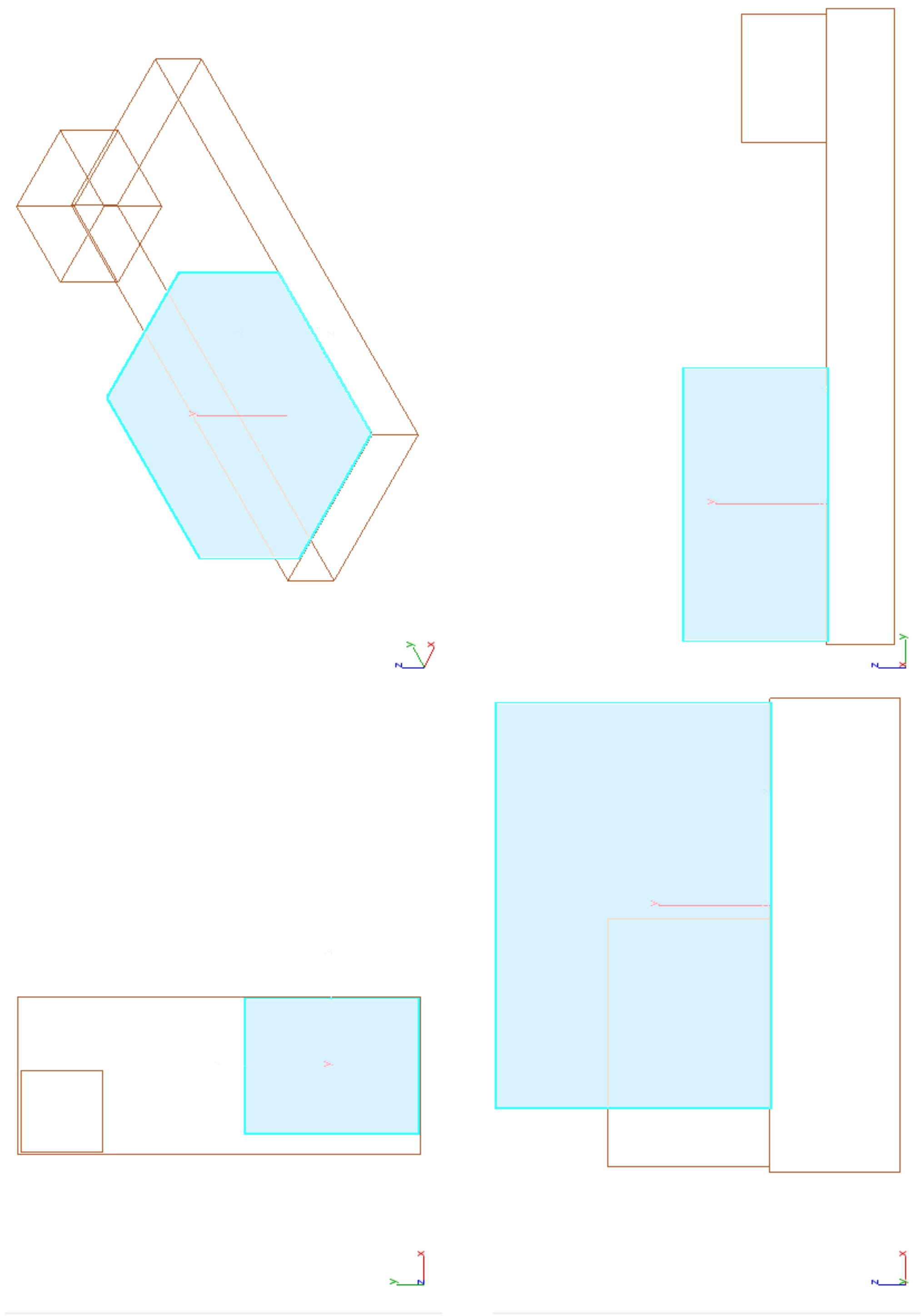


Figure 24: Orthographic render of Assembly Three, Amazon SPL Box

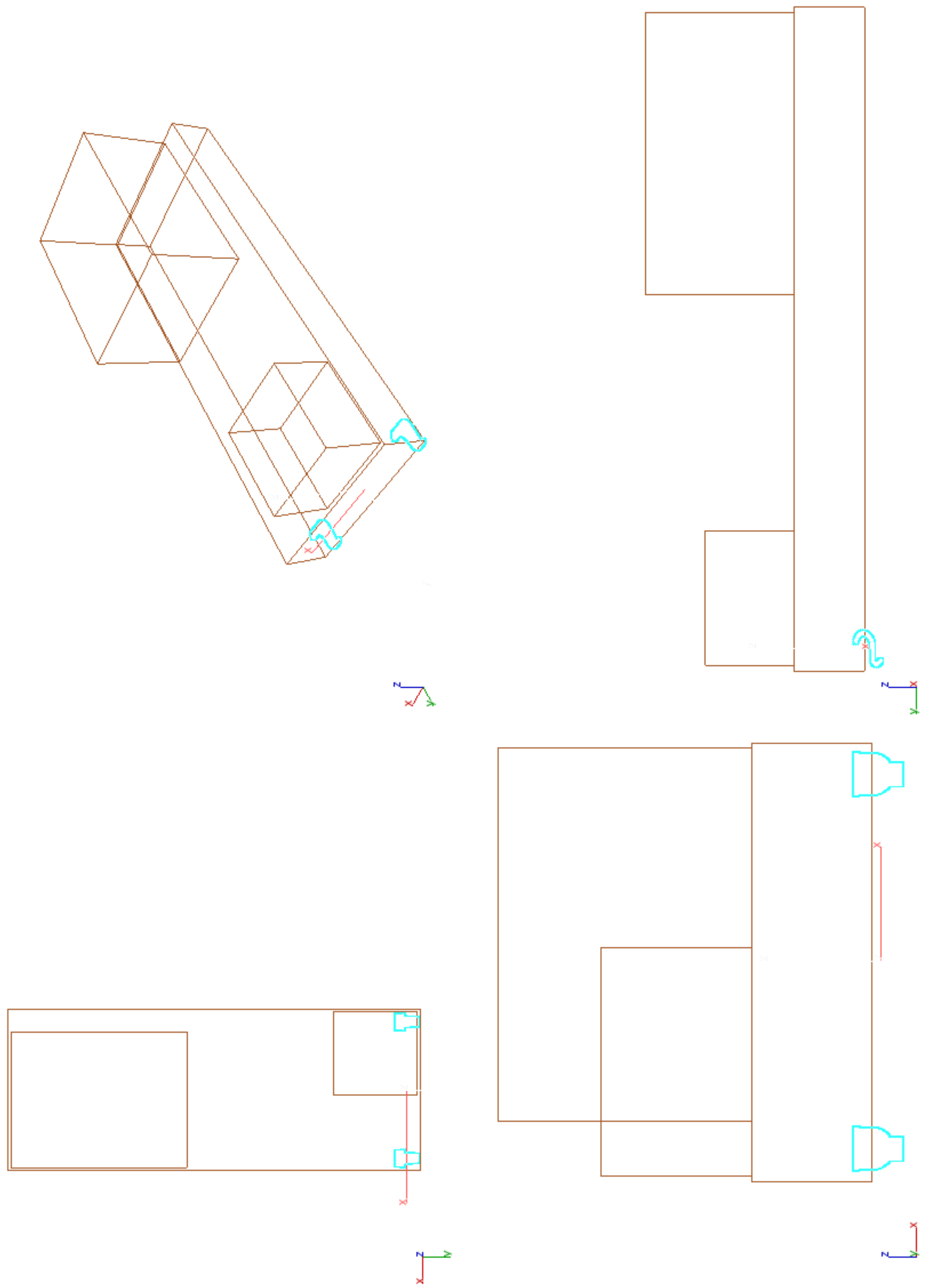


Figure 25: Orthographic render of Assembly Three, Support Hooks

Figure 26: Render of completed wing skin assembly with numbered channels

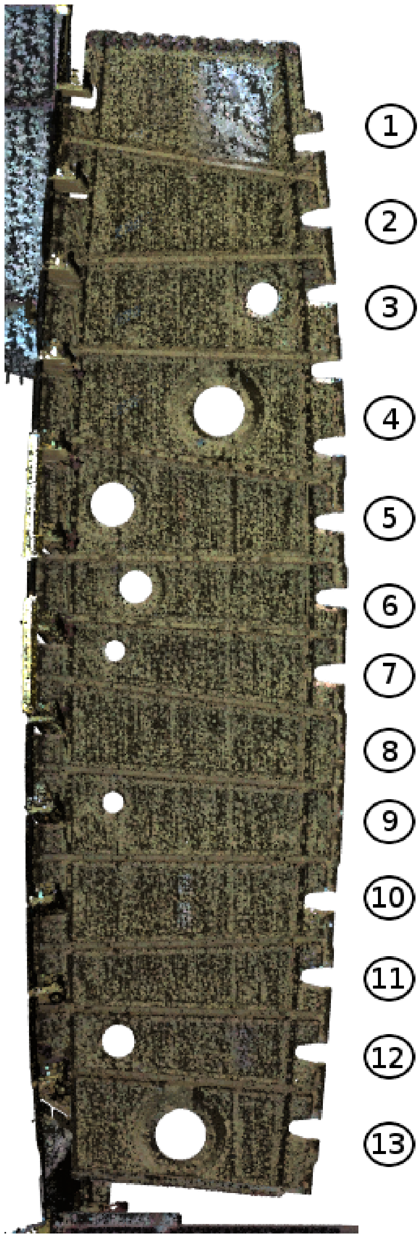


Figure 27: Render of left-hand side of wing skin stantion with numbered channels

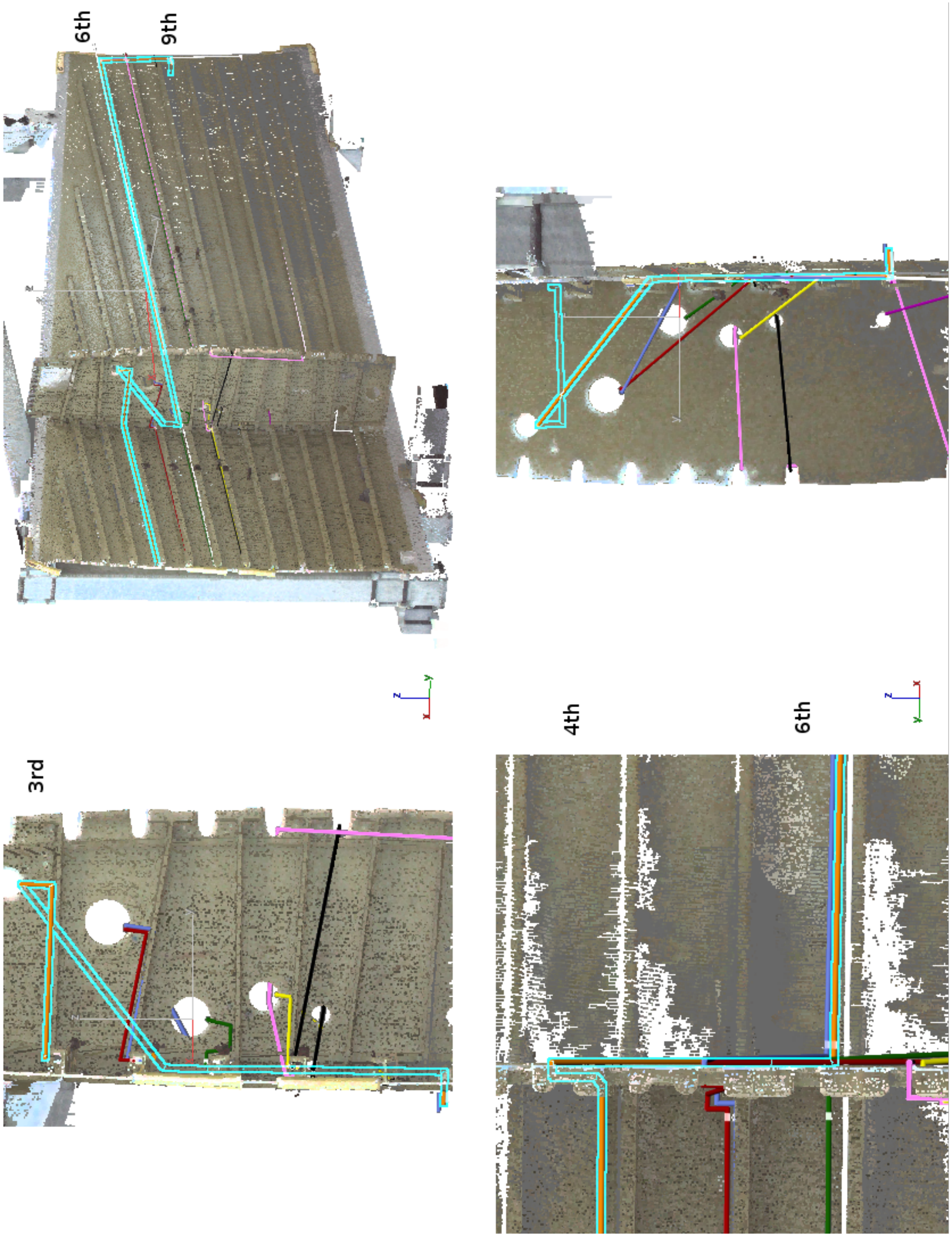


Figure 28: Orthographic renderings of the orange cable routing

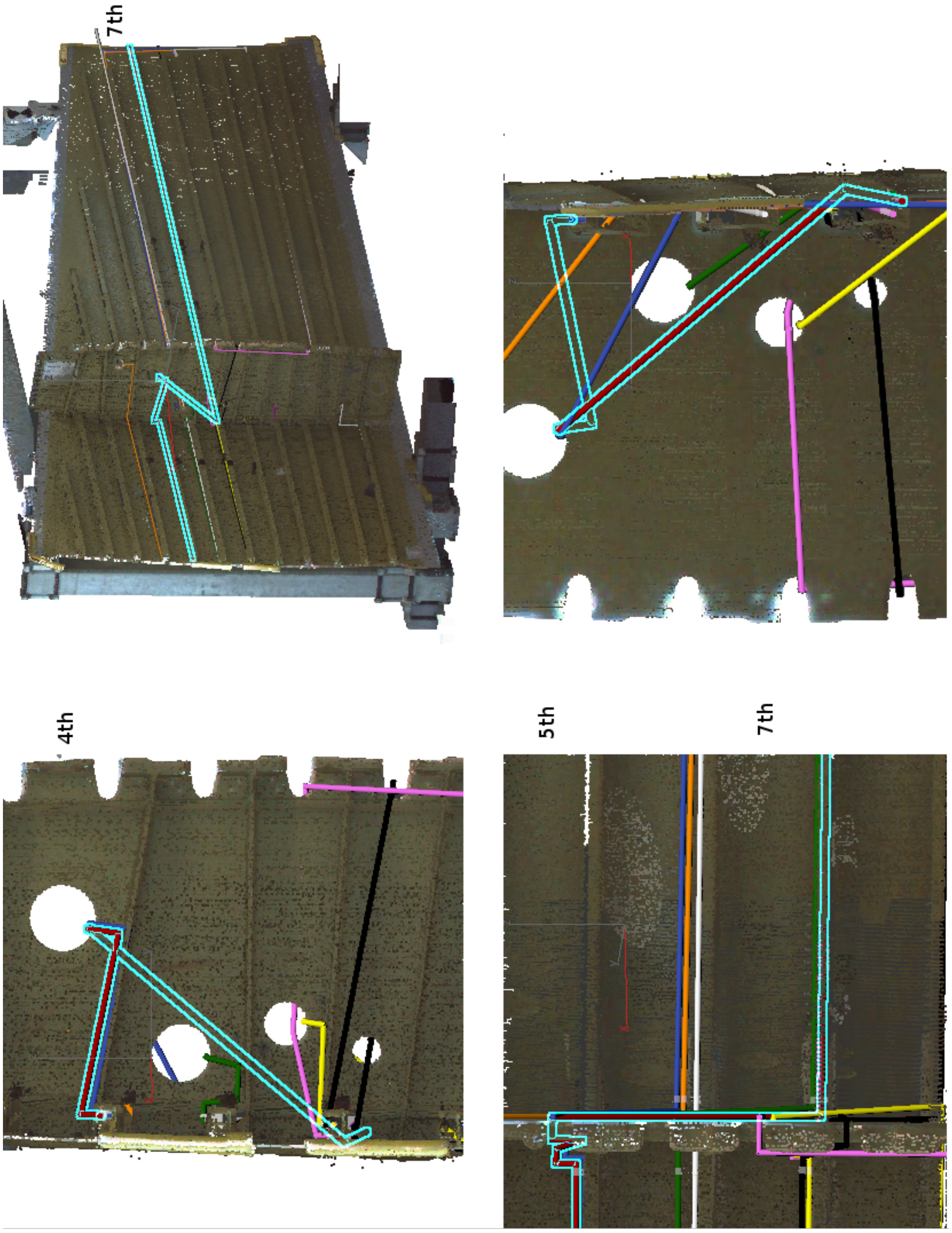


Figure 29: Orthographic renderings of the red cable routing

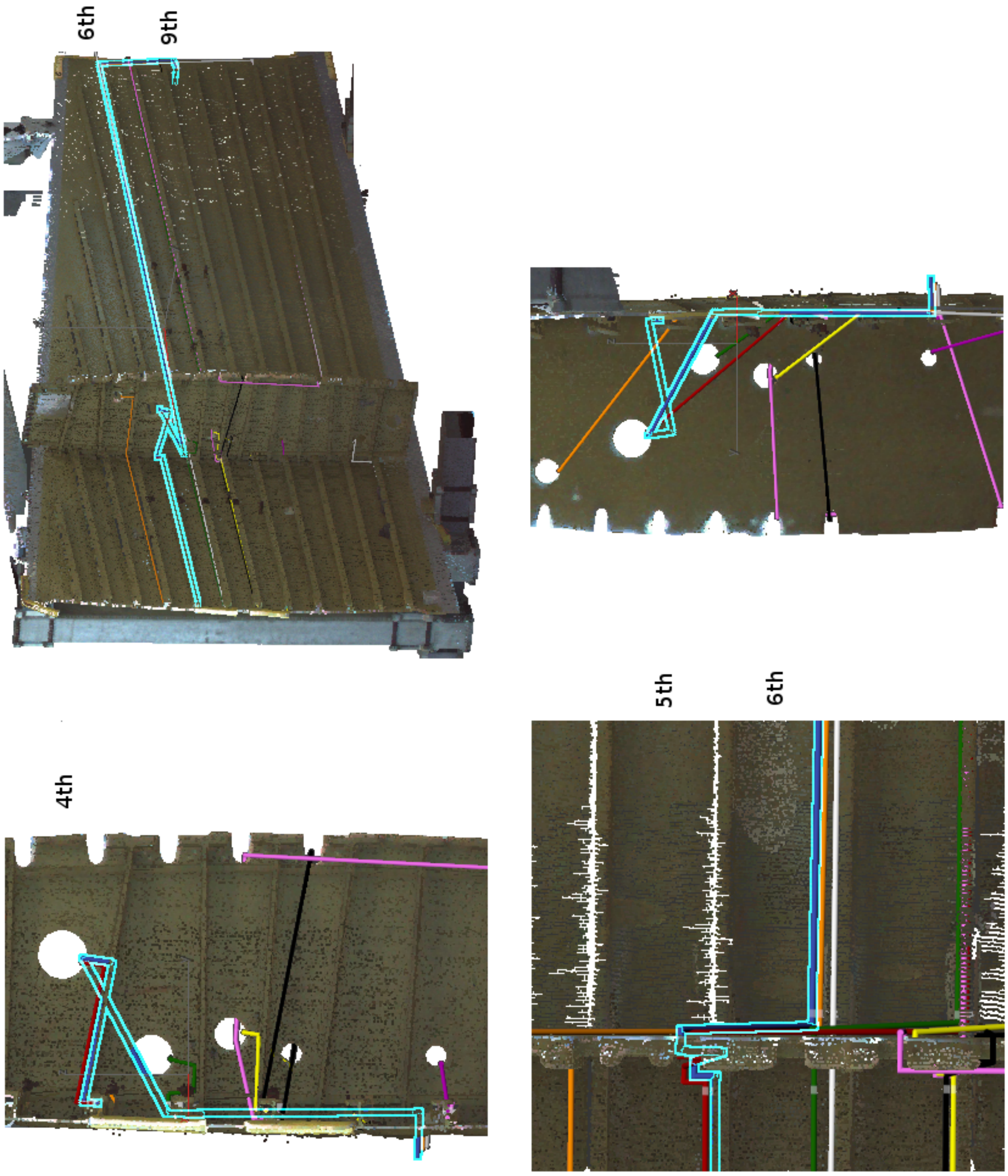


Figure 30: Orthographic renderings of the blue cable routing

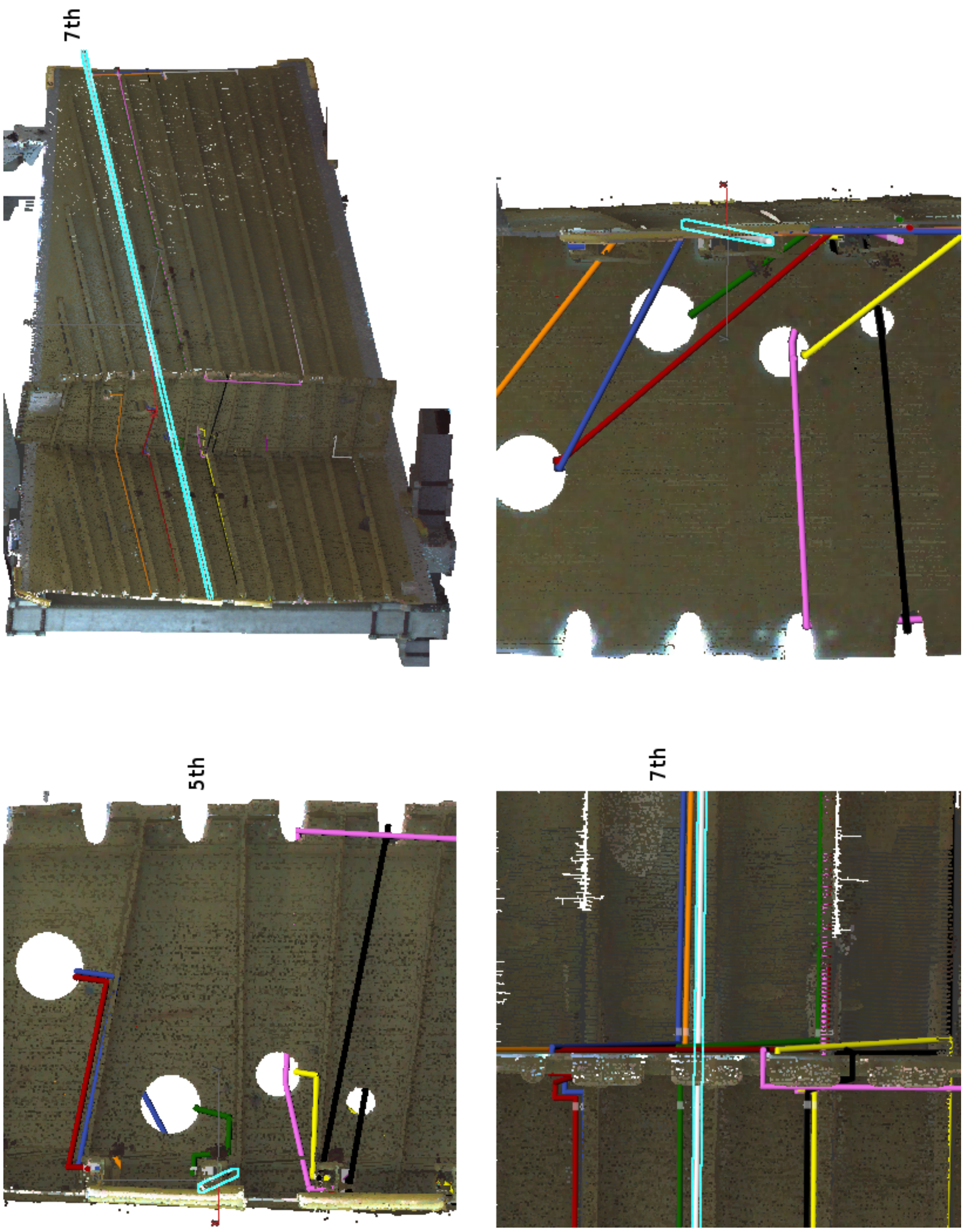


Figure 31: Orthographic renderings of the white cable routing

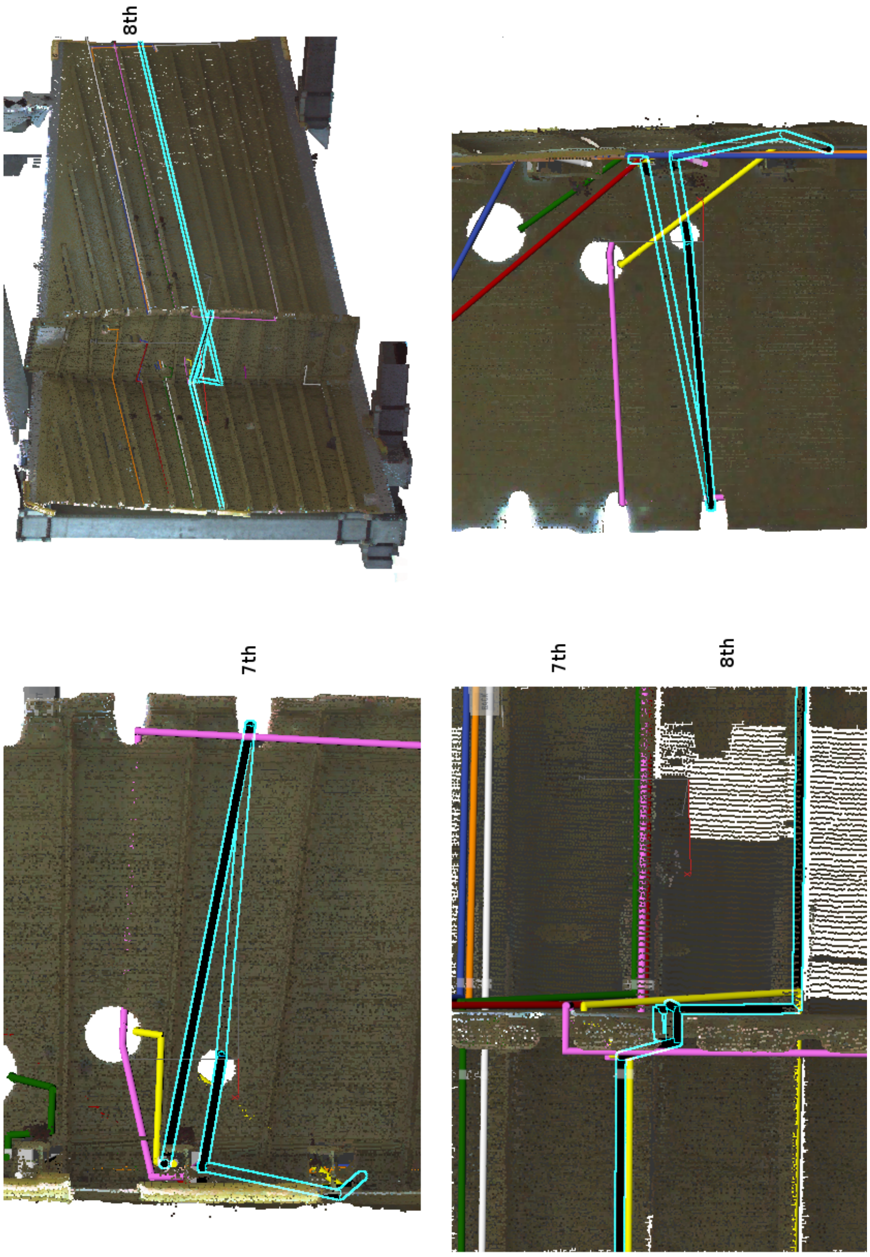


Figure 32: Orthographic renderings of the black cable routing

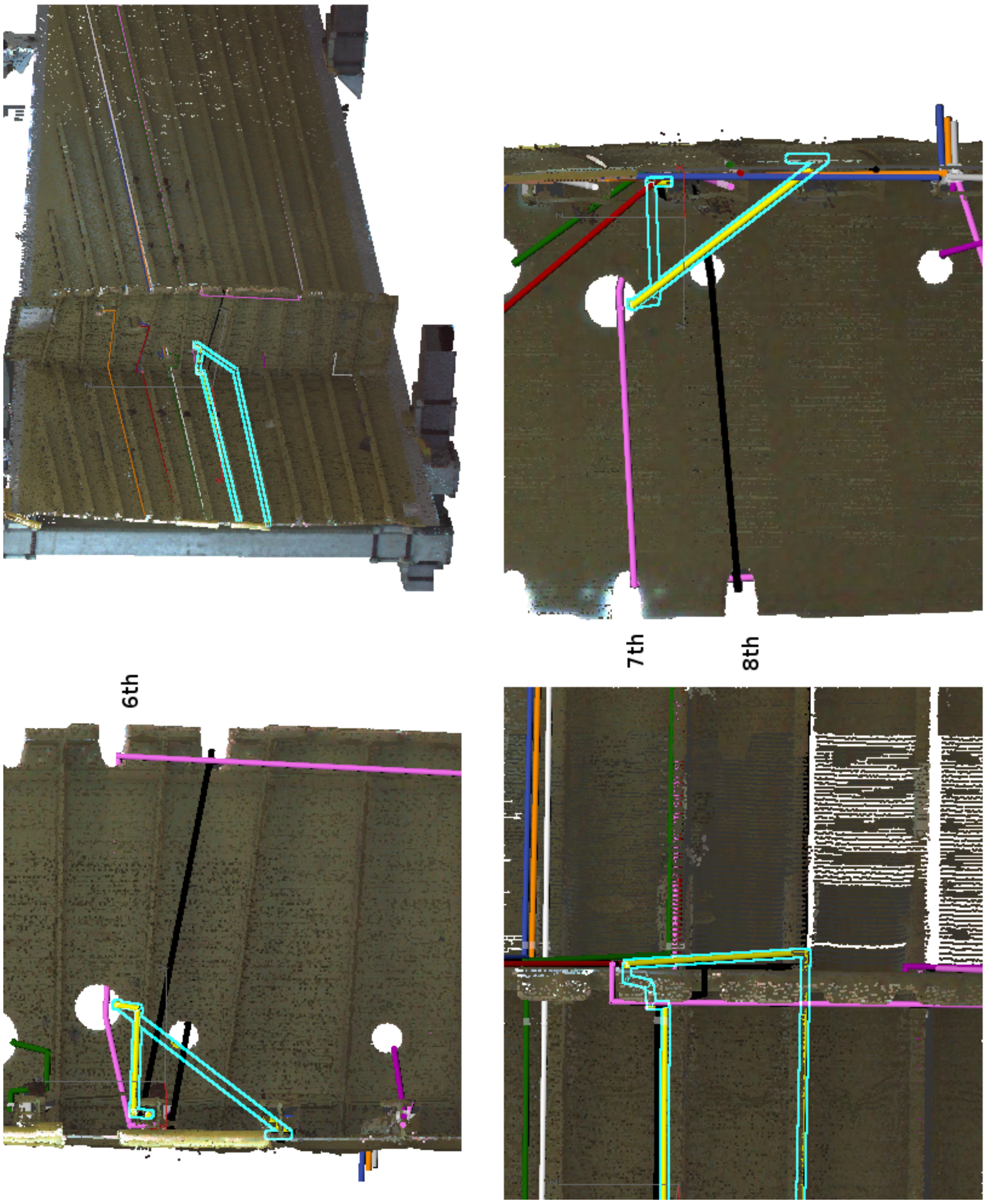


Figure 33: Orthographic renderings of the yellow cable routing

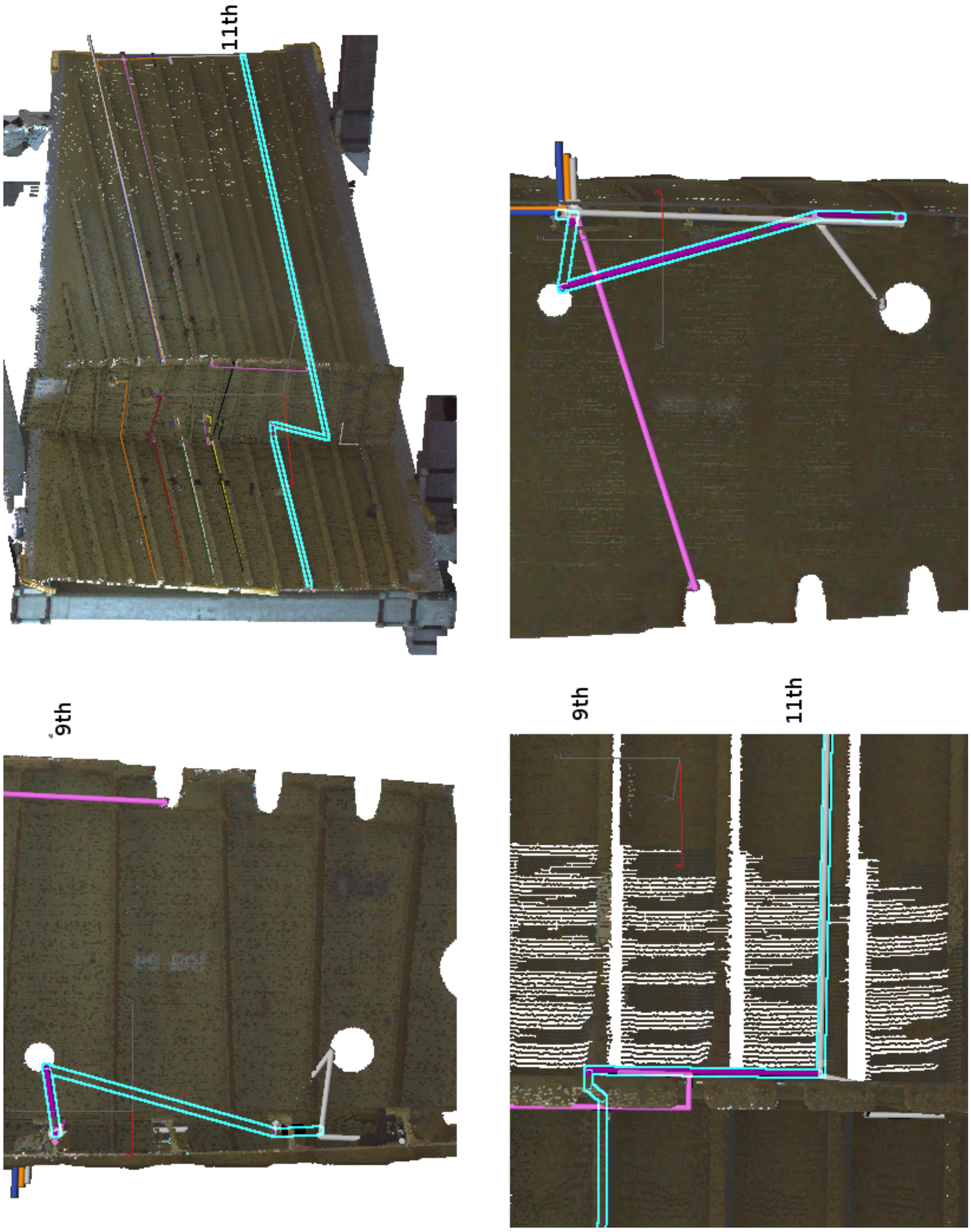


Figure 34: Orthographic renderings of the purple cable routing

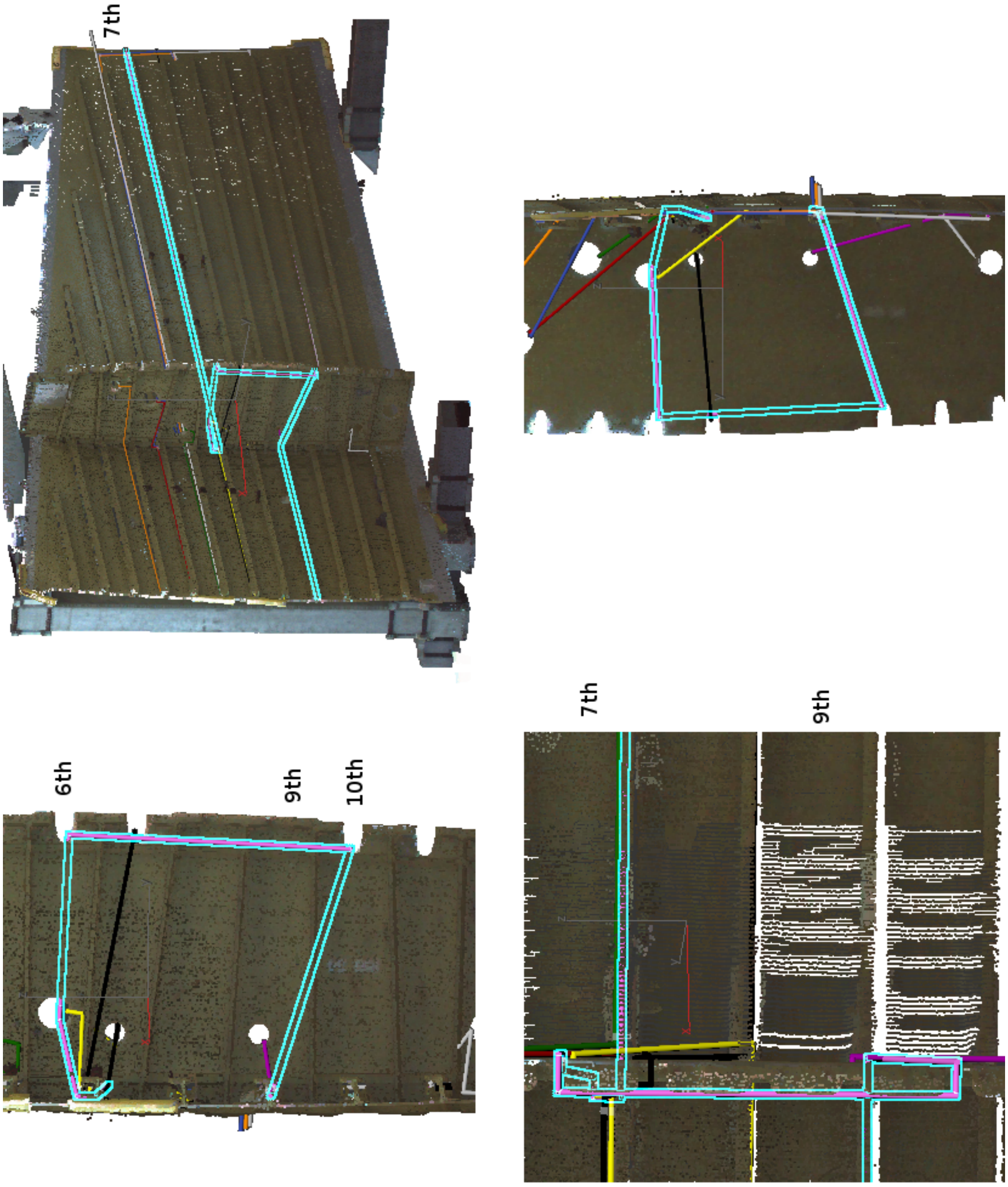


Figure 35: Orthographic renderings of the pink cable routing

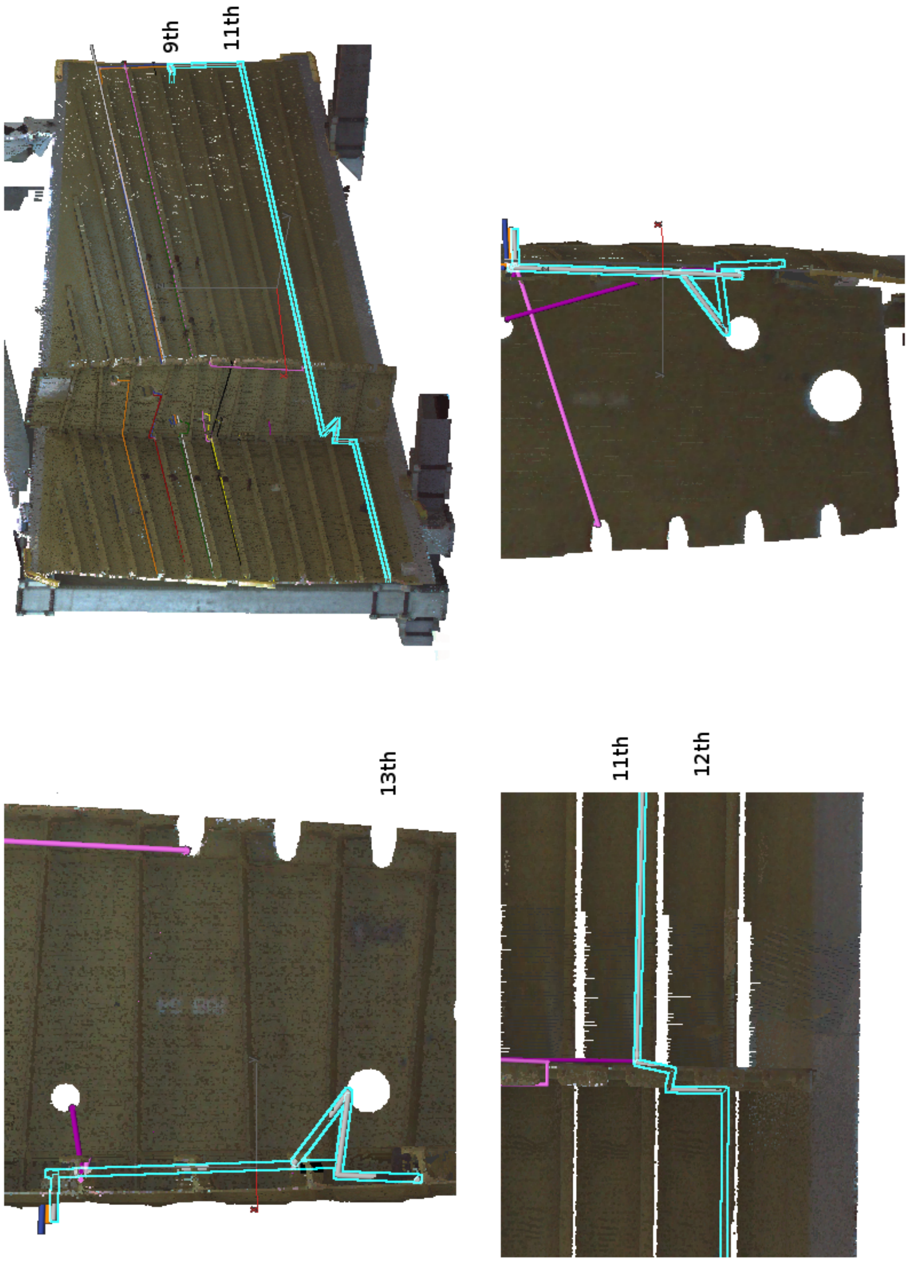


Figure 36: Orthographic renderings of the grey cable routing

Figure 37: Frontal render of wingskin with cable binding points‘

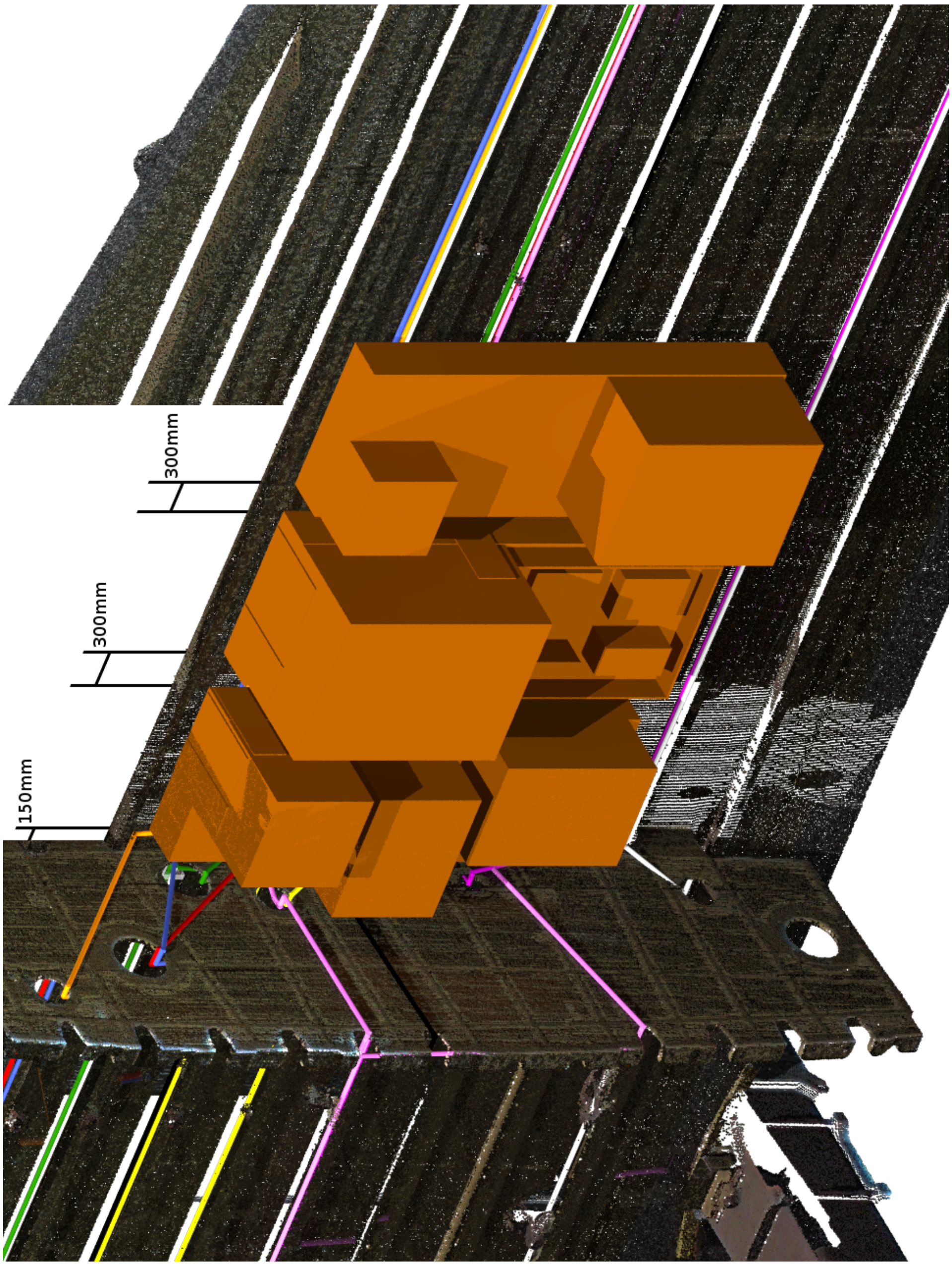


Figure 38: Render of Assemblies mounted on Wingskin

C.2 User Guide

UserGuide

A Bi-Directional data link for Virtual Environments

Author:
Stephen Reddish
s.reddish@namrc.co.uk

September 26, 2017

List of Contents

List of Figures

List of Tables

The following sections cover how to initiate the system, from booting the necessary hardware to in game controls.

1 Hardware

The order that the hardware is booted in is inconsequential. This list just serves as a reminder of what hardware must be switched on.

- Brix Kinectv2 servers 1 - 5
- Projector controller PCs
- Projectors
- Desktop Dell server
- Remote Virtual Environment (VE) desktop
- Android Tablet
- Wireless Router

2 Software

The following sections detail the software that must be running per machine, along with any oddities a user should be aware of.

2.1 Brix Kinectv2 servers

The Brix kinect servers are accessible through the NoMachine remote desktop softare on the Dekstop Dell server. Simply execute the software and connect to the correct IP. All the IPs of the system have been fixed, so these should remain a constant.

- Brix197 `192.168.0.197` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix181 `192.168.0.181` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix191 `192.168.0.191` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix198 `192.168.0.198` **UserName**: *DAACell* **Password**: *Factory2050*

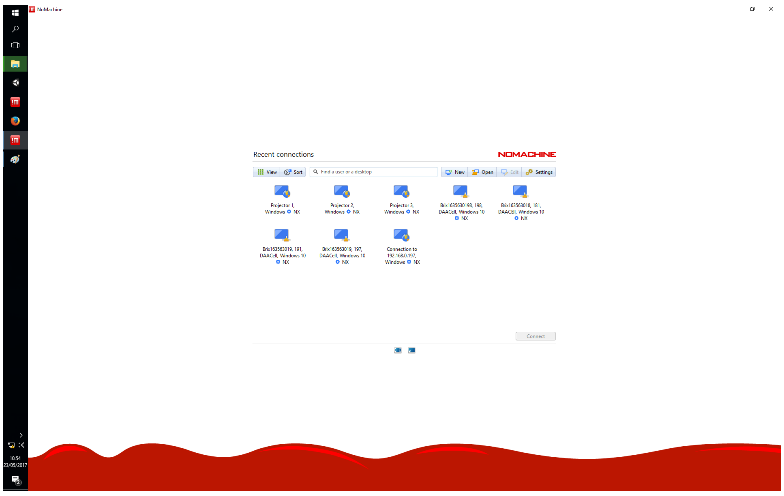


Figure 1: Screen capture of the NoMachine client interface

All machines require the same software run on them except Brix ...197 which also runs the Fusion Server. Run the KinectFusion sensor server, a shortcut is on the desktop.

2.1.1 Brix197

Brix197 acts as the kinect Fusion Server, so you will need to execute the the Fusion Server program as well. A shortcut for this is on the desktop. The software itself is currently calibrated for the current Kinectv2 set up.

The sensors should appear on the right hand side of the the Fusion server window, if they do not just physically disconnect the relevant kinectv2 from the USB port and re-connect it. Sensors can be added manually by providing the software with the required IP address. To reduce netwrok traffic ensure that the sensor servers are only sending the skeletal data, and not the depth or RGB data. This is done by unchecking the green boxes under the sensor IP in its listing.

If you move any of the sensors you will need to re-calibrate the software. This is done by running the kinect calibration wizard within the Fusion Server program. Followed by making T-poses in front of each Kinect until calibration is complete. The calibrations can be saved along with the hole sensor setup. Futher more the coordinate space can be universally rotated or translated within the Fusion Server itself.

2.2 Projector controller PCs

These machines are accesible by using the remote desktop program NoMachine from the destop server machine.

- PWOperator1 `192.168.0.101` **UserName**: *PWOperator* **Password**: *pwoperator*
- PWOperator2 `192.168.0.102` **UserName**: *PWOperator* **Password**: *pwoperator*
- PWOperator3 `192.168.0.103` **UserName**: *PWOperator* **Password**: *pwoperator*

The computers always boot with the inherent projector software so when you first login you will be greeted with a complete black screen. Simply press the windows key and then close the projector softare to return to normal functionality.

2.3 Fusion XV

Fusion XV is the unity executable must be run on the PWOperator1 - 3, the Dell Desktop server and any remote VE running desktops. The android apk must be run on any Android device that you wish to connect. It will need access to the network

Once executed the program will show you the layout of the virtual cell through a rotating camera. In the top left is the High Level API (HLAPI) GUI overlay.
If you are running the instance on the server, click the host button. If you are running as a client, ie the projectors, any VE clients or an Android application then click within the box to give an IP address to connect as client to.

The IP is the one of the server machine `192.168.0.101`. The click the client button next to it.

2.3.1 In Game Controls

Once in game there are a variety of controls dependant on what platform you are using. The on screen GUI works in the same way on every platform. Futhermore the same objects in game can be interacted with on all platforms, although the touch controls on Android leave a lot to be desired.

Desktop
Whithin the desktop environment the player character can be controlled by the game standard **WASD** keys while **mouse button two** is depressed. Additionally keyboard button **E** increases the elevation and **Q** decreases the elevation fo the player character. The mouse controls the players rotation. The **mouse button one** will allow the user to pick up and while the button is depressed hold objects. Dragging the boxes inline with their corresponding position will cause them to change colour to green and move to their specified location.

The work instructions can be turned on and off by pressing the keyboard button **U**.

- Keyboard button **WASD** Action: Moves the player character through the scene
- Keyboard button **E** Action: Increases the elevation of the player character
- Keyboard button **Q** Action: Decreases the elevation of the player character
- Keyboard button **U** Action: Toggles WorkInstructions on/off

Android
The GUI layer is available on all deployments of the Unity software. It contains a menu button in the top right, which will cause a panel to slide out with more buttons. The buttons control the Voice over Internet Protocol (VoIP), the location or camera view of the user, and the work instructions shown to every user on the same network instance.

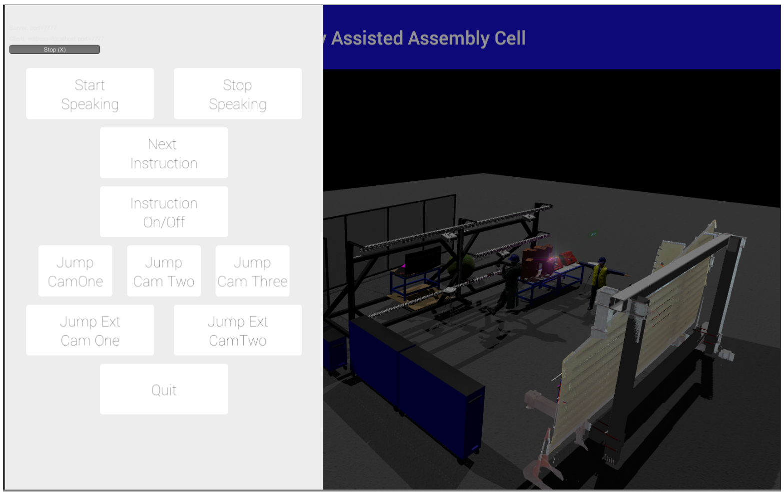


Figure 2: Android application GUI

Alongside the GUI buttons you can use a two finger drag to rotate the camera angle. The sensitivity on this is very high, which is why they GUI buttons exist to teleport the user to useful locations.

Projectors

For the projectors there are a few shortcuts that will prove useful. The first of which is the keyboard button G which turns the GUI off so that it no longer inhibits the projection of the scene. The keyboard button **P** which switch the view cam cull on for the camera being used. This reduces the camera sight to only what is part of the work instructions or a grabbable object. Finally holding **mouse button two** and pressing keyboard button **1/2/3** will switch the users view to the correct location and rotation for each projector 1/2/3.

- Keyboard button **G** Action: Toggles GUI on/off
- Keyboard button **P** Action: Switches camera to Projector mode, culling unnecessary objects from render scene

Furthermore I strongly recommend using the GUI to "*stop talking*" as the projector don't have a built in mic. This reduces network traffic.

Virtual Environment Clients

Within the VE motion is controlled by the user. Their is a teleportation function which is activated by using either trigger switch on the VIVE controllers. Depressing the trigger will bring up a laser, and pulling on the trigger will cause the player to teleport to that location. Objects can be picked up by using the controllers side mounted grab buttons. Once both are depressed the object will attach to the players controller until they are pressed again. Pressing the dial pad will bring up an attached menu. The top button will toggle the voice transmission on or off. The bottom button will toggle the workin instructions GUI on or off. The right button will progress the work instructions forward one step, whereas the left button will move the instructions back one step.

Wifi Router

The WiFi router is accessible through any web browser on a machine connected to the network. The router software is used to fix the IPs for all the machines except the Android tablet and VE client machines

- WiFi Router `192.168.0.254` **UserName**: *admin* **Password**: *DAACellF2050*

3 System Overview

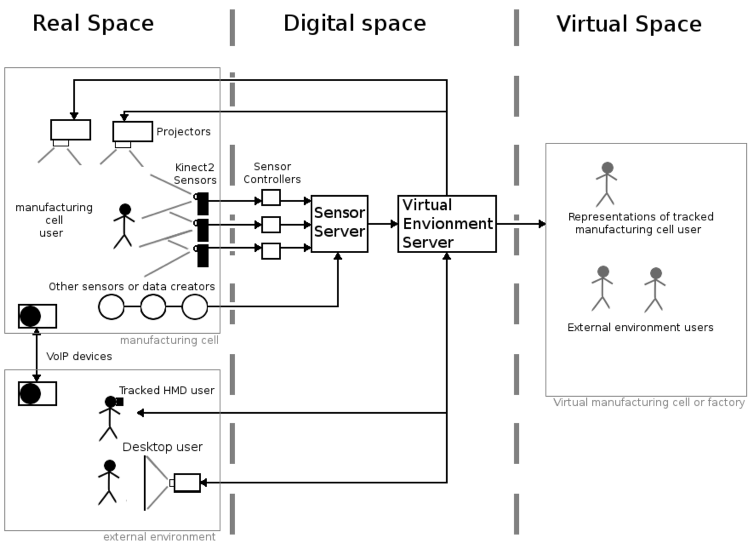


Figure 3: Project Diagram

A IP tables

- Brix197 `192.168.0.197` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix181 `192.168.0.181` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix191 `192.168.0.191` **UserName**: *DAACell* **Password**: *Factory2050*
- Brix198 `192.168.0.198` **UserName**: *DAACell* **Password**: *Factory2050*

- PWOperator1 `192.168.0.101` **UserName**:*PWOperator* **Password**: *pwoperator*
- PWOperator2 `192.168.0.102` **UserName**:*PWOperator* **Password**: *pwoperator*
- PWOperator3 `192.168.0.103` **UserName**:*PWOperator* **Password**: *pwoperator*

- DellHQ7431 `192.168.0.101` **UserName**: **Password**:

- Router `192.168.0.254` **UserName**: *admin* **Password**: *DAACellF2050*

B HotKey table

- Keyboard button **WASD** Action: Moves the player character through the scene
- Keyboard button **E** Action: Increases the elevation of the player character
- Keyboard button **Q** Action: Decreases the elevation of the player character
- Keyboard button **U** Action: Toggles WorkInstructions on/off
- Keyboard button **G** Action: Toggles GUI on/off
- Keyboard button **P** Action: Switches camera to Projector mode, culling unnecessary objects from render scene

C Unity Scripts

This is the RayCastPickUp.cs Script

```
using UnityEngine;
using UnityEngine.Networking;
using UnityEngine.VR;
using Dissonance.VAD;

public class PlayerSetup : NetworkBehaviour {

    public Camera playerCamera;
    [SerializeField] Behaviour[] componentsToDisable;
    [SerializeField] Behaviour[] disableIfNotVR;
    [SerializeField] GameObject VRTK;
    Camera sceneCamera;
    [SerializeField] GameObject[] componentsToSpawn;
    public GameObject PlayerCapsule;
    bool cullingMaskOnOff;

    private void Start()
    {
        if(!isLocalPlayer)
        {
            for (int i = 0; i < componentsToDisable.Length; i++)
            {
                componentsToDisable[i].enabled = false;
            }
        }
        else
        {
            sceneCamera = Camera.main;
            if (sceneCamera != null)
            {
                sceneCamera.gameObject.SetActive(false);
            }
        }

        if (isLocalPlayer && VRDevice.isPresent) //Only enable VRTK if
    VRDevice is preset, and only on the local player.
        {
            VRTK.SetActive(true);
        }

        if (isLocalPlayer && !VRDevice.isPresent) //only disable if VRDevice
     is not present, and only on the local player.
        {
            for (int i = 0; i < disableIfNotVR.Length; i++)
            {
                disableIfNotVR[i].enabled = false;
            }
```

```
        }
    }

    private void OnDisable()
    {
        if (sceneCamera != null)
        {
            sceneCamera.gameObject.SetActive(true);
        }
    }

    private void Awake()
    {
        for (int i = 0; i < componentsToSpawn.Length; i++)
        {
            Vector3 spawnPosition = new Vector3(0, 0, 0);
//              Quaternion spawnRotation = Quaternion.Euler(0, 0, 0);
            GameObject spawnObject = (GameObject)Instantiate(
    componentsToSpawn[i], spawnPosition, transform.rotation);
            spawnObject.SetActive(true);
            Debug.Log("spawnObject is " + spawnObject.name);
            NetworkServer.Spawn(spawnObject);
        }
    }

    public void Update()
    {
        if (Input.GetKeyDown(KeyCode.P))
        {
            cullingMaskOnOff = !cullingMaskOnOff;

            if(cullingMaskOnOff == true)
            {
                playerCamera.cullingMask = (1 << LayerMask.NameToLayer("
    ViewCamCull")) | (1 << LayerMask.NameToLayer("PickUpLayer"));
            }
            else
            {
                playerCamera.cullingMask = -1;
            }
        }
    }

    public override void OnStartLocalPlayer()
    {
        PlayerCapsule.GetComponent<MeshRenderer>().material.color = Color.
    blue;
```

```
    }
}
```

Listing 1: C++ RayCastPickUp Code

```
using UnityEngine;
using System.Collections;
using UnityEngine.Networking;

public class RayCastPickUp : NetworkBehaviour
{
    [SerializeField]
    private float distance = 1.0f;
    [SerializeField]
    private Camera playerCamera;
    private RaycastHit hit;
    [SyncVar]
    private GameObject objectID;
    private NetworkIdentity objNetId;


    private int layerMask = 1 << 9; //pickUpLayer is 9

    void Update()
    {
        if (isLocalPlayer)
        {
            CheckIfMoving();
        }
    }

    void CheckIfMoving()
    {
        if (!isLocalPlayer)
        {
            return;
        }
        Ray ray = playerCamera.ScreenPointToRay(Input.mousePosition);

        if (Input.GetMouseButton(0))
        {

            if (Physics.Raycast(ray.origin, ray.direction, out hit, Mathf.
  Infinity, layerMask))
            {
                objectID = GameObject.Find(hit.transform.name);
                        // this gets the object that is hit
                CmdPickUp(objectID);
```

```
            }
        }
    }


/*    [ClientRpc]
    void RpcPickUp(GameObject objectID, Vector3 objPosition, Quaternion objRotation)
    {
        objectID.transform.position = objPosition;
        objectID.transform.rotation = objRotation;
    }
*/
    [Command]
    void CmdPickUp(GameObject objectID)
    {
        objNetId = objectID.GetComponent<NetworkIdentity>();        // get the object's network ID
        objNetId.AssignClientAuthority(connectionToClient);    // assign authority to the player who is changing the color

        Vector3 mousePosition = new Vector3(Input.mousePosition.x, Input.mousePosition.y, distance);
        Vector3 objPosition = playerCamera.ScreenToWorldPoint(mousePosition);
        Quaternion objRotation = playerCamera.transform.rotation;

        objectID.transform.position = objPosition;
        objectID.transform.rotation = objRotation;
        //        RpcPickUp(objectID, objPosition, objRotation);                // use a Client RPC function to "paint" the object on all clients

        this.GetComponent<Renderer>().material.color = Color.clear; // finally change color to transparent
        objNetId.RemoveClientAuthority(connectionToClient);    // remove the authority from the player who changed the color
    }
}
```

Listing 2: C++ RayCastPickUp Code

C.3 User Testing

User Testing Feedback

A Bi-Directional data link for Virtual Environments

Author:

Stephen Reddish

s.reddish@namrc.co.uk

May 23, 2017

1 Questions on the Environment

Please rate your experience of the system in the following area.

1. Please state which Environments you used?
   ◯ Android ◯ Desktop ◯ Virtual Environment

2. How easy was the application to use?
   ◯ Very easy ◯ Easy ◯ Functional ◯ Difficult ◯ Very difficult

3. How intuitive did you find the in application controls?
   ◯ Very intuitive ◯ Intuitive ◯ Functional ◯ Unintuitive ◯ Very unintuitive

4. Was it easy to see what was taking place within the Virtual Environments (VEs)?
   ◯ Yes ◯ No

5. Were there enough controls available to you?
   ◯ Yes ◯ No
   If no what other controls would you have like to see?

How well did the application represent the.....

6. The wingskin?
   ◯ Very well ◯ Well ◯ Functional ◯ Poorly ◯ Very poorly

7. The tracked user?
   ◯ Very well ◯ Well ◯ Functional ◯ Poorly ◯ Very poorly

8. The work instructions?
   ◯ Very well ◯ Well ◯ Functional ◯ Poorly ◯ Very poorly

9. The other users?
   ◯ Very well ◯ Well ◯ Functional ◯ Poorly ◯ Very poorly

Please rate the following means of communication:

10. Voice over Internet Protocol (VoIP)?
    ◯ Very effective ◯ Effective ◯ Functional ◯ Less than functional ◯ Inoperable

11. On wing projections?
    ◯ Very effective ◯ Effective ◯ Functional ◯ Less than functional ◯ Inoperable

12. Shared control of the work instructions?
    ◯ Very effective ◯ Effective ◯ Functional ◯ Less than functional ◯ Inoperable

User Feedback

13. What issues or difficulties (if any) did you encounter while using the system?

14. What improvements and/or changes would you want to see in future iterations of the system?

15. Where and how do you see this system being used within a manufacturing environment?

16. What other use cases could you envisage for the system?

User Feedback

17. Do you think this is a viable method for delivering remote assistance?

18. Is the level of assistance offered through the VE sufficient?

19. Do you consider any aspects of the project to be superfluous?

20. Any further comments about the system?

APPENDICES D

Code

Contained below is a selection of scripts written and used in Unity3D. The list is anything but exhaustive, but is used as examples to demonstrate how C++ and JavaScript are used within Unity3D.

D.1 DBUnityHelper.cs

```
//---------------------------------------------------------------
// DynamoDB Helper
// Copyright Â© 2014-2015 OuijaPaw Games LLC
//---------------------------------------------------------------

using System.Collections.Generic;
using Amazon;
using UnityEngine;

namespace DDBHelper
{
    public class DBUnityHelper : MonoBehaviour
    {
        public enum ConnectType : int
        {
            DDB_Cognito,
            DDB_Normal,
            DDB_Local
        }

        public static DBUnityHelper Instance;
        public static bool SHOW_DEBUG = true;

        private static Queue<SendMessageContext> _queuedMessages = new Queue<SendMessageContext>();
        private static Queue<SendCoroutineContext> _queuedCoroutines = new Queue<SendCoroutineContext
>();

        public static DBConnect dbConnect { get; private set; }
#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ General ------")]
#endif
```

```
        public bool showDebugMessages = true;

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ Connection ------")]
#endif
        public ConnectType connectType;
        public DBConnect.region regionEndPoint;
        public string IdentityPoolId = "Use for Cognito ID";
        public string AccessKeyID = "Use for DDBLocal / remote DDB";
        public string SecretKeyID = "Use for non-cognito remote DDB";

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ DynamoDB Local ------")]
#endif
        public int localPort = 8000;

#if !UNITY_4_0 && !UNITY_4_1 && !UNITY_4_2 && !UNITY_4_3
        [Header("------ Proxy ------")]
#endif
        public bool useProxyForConnection = false;
        public string proxyHostName = "";
        public int proxyPortNumber = 0;
        public string proxyUserName = "";
        public string proxyPassword = "";

        [SerializeField]
        public RegionEndpoint reg;

        public System.Net.NetworkCredential proxyCredentials { get; private set; }

        /// <summary>
        /// Initializes everything on Awake. Since Awake order is undetermined, do not try to load anything
        /// from the DBWorker until Start() method or later.
        /// </summary>
        void Awake()
        {
            // Keeping a single instance of this class as reference
            if (Instance)
                Destroy(Instance);
            Instance = this;

            SHOW_DEBUG = showDebugMessages;

            Amazon.UnityInitializer.AttachToGameObject(this.gameObject);

            if (useProxyForConnection)
                proxyCredentials = new System.Net.NetworkCredential(proxyUserName, proxyPassword);

            dbConnect = new DBConnect();

            switch (connectType)
            {
                default:
```

```
                case ConnectType.DDB_Cognito:
                    dbConnect.InitRemoteConnection( IdentityPoolId );
                    break;
                case ConnectType.DDB_Normal:
                    dbConnect.InitRemoteConnection(AccessKeyID, SecretKeyID);
                    break;
                case ConnectType.DDB_Local:
                    dbConnect.InitLocalConnection( localPort );
                    break;
            }

            // creates the worker thread/connection/context/etc.
            DBWorker.InitializeForUnity(dbConnect, useProxyForConnection, (connectType == ConnectType.
DDB_Local));
        }

        /// <summary>
        /// Used for registering a gameobject that will use SendMessage to invoke a method
        /// </summary>
        /// <param name="context"></param>
        public static void Register(SendMessageContext context)
        {
            lock (_queuedMessages)
            {
                _queuedMessages.Enqueue(context);
            }
        }

        /// <summary>
        /// Used for registering a MonoBehaviour that will use StartCoroutine to invoke an IEnumerator
        /// </summary>
        /// <param name="context"></param>
        public static void Register(SendCoroutineContext context)
        {
            lock (_queuedCoroutines)
            {
                _queuedCoroutines.Enqueue(context);
            }
        }

        /// <summary>
        /// This is dependent upon framerate. Move to fixedupdate if there are performance issues.
        /// If you need to update graphics or something of the sort, move to lateupdate.
        /// </summary>
        private void Update()
        {
            while (_queuedMessages.Count > 0)
            {
                SendMessageContext context = null;
                lock (_queuedMessages)
                {
                    context = _queuedMessages.Dequeue();
                }

```

```
                // rare case could have been destroyed
                if (context.Target)
                    context.Target.SendMessage(context.MethodName, context.Value, context.Options);
            }

            while (_queuedCoroutines.Count > 0)
            {
                SendCoroutineContext context = null;
                lock (_queuedCoroutines)
                {
                    context = _queuedCoroutines.Dequeue();
                }

                // rare case could have been destroyed
                if (context.Target)
                    context.Target.StartCoroutine(context.MethodName, context.Value);
            }
        }
    }

    /// <summary>
    /// Used for storing reference to GameObject and methodname used for SendMessage
    /// </summary>
    public class SendMessageContext
    {
        public GameObject Target;
        public string MethodName;
        public object Value;
        public SendMessageOptions Options = SendMessageOptions.RequireReceiver;

        public SendMessageContext(GameObject target, string methodName, object value, SendMessageOptions
options)
        {
            this.Target = target;
            this.MethodName = methodName;
            this.Value = value;
            this.Options = options;
        }
    }

    /// <summary>
    /// Used for storing reference to MonoBehaviour and methodname used for Coroutine
    /// Changed the target to a monobehaviour so they are easily differentiated
    /// </summary>
    public class SendCoroutineContext
    {
        public MonoBehaviour Target;
        public string MethodName;
        public object Value;

        public SendCoroutineContext(MonoBehaviour target, string methodName, object value)
        {
            this.Target = target;
            this.MethodName = methodName;
```

```
                this.Value = value;
            }
        }
}
```

D.2 DBUnitySFrame.cs

```
using System;
using System.Collections.Generic;
using Amazon.DynamoDBv2;
using Amazon.DynamoDBv2.Model;
using Amazon.Runtime;
using UnityEngine;

namespace DDBHelper
{
    public class DBUnitySFrame : MonoBehaviour
    {
        public string AWSTableName = "unity";
        public bool AWSConnection = true;
        public bool ScanTable = true;

        void ReScanTable()
        {
            DBWorker.Instance.ScanTable(AWSTableName, CapacityType.INDEXES);
        }

        void Start ()
        {
            if (AWSConnection)
                DBConnect.TestAWSConnection(AWSTableName);

            if (ScanTable)
            {
                InvokeRepeating("ReScanTable",2,15);
            }
        }

public void OnClick()
        {
            DBWorker.Instance.ScanTable(AWSTableName, CapacityType.INDEXES);
        }

    }
}
```

D.3 InternalSensorTag.cs

```
ï»¿using UnityEngine;
using System.Collections;

namespace DDBHelper
{
    public class InternalSensorTag : MonoBehaviour
    {
        Transform labelHolder;

        void Awake()
        {
            labelHolder = transform.Find("LabelHolder");
        }

        // Use this for initialization
        void Update()
        {
            var dateTextMesh = labelHolder.Find("Date").GetComponent<TextMesh>();
            dateTextMesh.text = AssetQuery.internalDate;
            var lightTextMesh = labelHolder.Find("Light").GetComponent<TextMesh>();
            lightTextMesh.text = "Light " + AssetQuery.internalLight + "Lux";
            var humidityTextMesh = labelHolder.Find("Humidity").GetComponent<TextMesh>();
            humidityTextMesh.text = "Humidity " + AssetQuery.internalHumidity + "%";
            var tempTextMesh = labelHolder.Find("Temp").GetComponent<TextMesh>();
            tempTextMesh.text = "Temp " + AssetQuery.internalTemp + "C";
        }
    }
}
```

D.4 AssetQuery.cs

```
ï»¿using System;
using System.Collections.Generic;
using Amazon.DynamoDBv2;
using Amazon.DynamoDBv2.Model;
using Amazon.Runtime;
using UnityEngine;

namespace DDBHelper
{
    // this class contains the functions that allocate the sorted scanned data from DBWorker.cs' ScanTable function
    public class AssetQuery
    {
        public static string internalDate, internalTemp, internalLight, internalHumidity;
        public static string externalDate, externalTemp, externalLight, externalHumidity;

        /// <summary>
        /// Used for sorting to see each type of value and the data of that value
        /// in an AttibuteValue pair
        /// </summary>
        /// <param name="dict"></param>
        public static void PrintDictValuesInternal(Dictionary<string, AttributeValue> dict)
        {
            foreach (var keyValuePair in dict)
            {
                //                Debug.Log("KeyValuePair #" + counter++ +
                //                          " K=" + keyValuePair.Key +
                //                          " S=" + keyValuePair.Value.S);
                if (keyValuePair.Key == "date")
                { internalDate = keyValuePair.Value.S; }
                if (keyValuePair.Key == "temp")
                { internalTemp = keyValuePair.Value.S; }
                if (keyValuePair.Key == "light")
                { internalLight = keyValuePair.Value.S; }
                if (keyValuePair.Key == "humidity")
                { internalHumidity = keyValuePair.Value.S; }
            }
            Debug.Log("Values for internal_sensors are internal date =" + internalDate + " InternalTemp=" + internalTemp + " InternalLight= " + internalLight + " InternalHumidity= " + internalHumidity);
        }

        public static void PrintDictValuesExternal(Dictionary<string, AttributeValue> dict)
        {
            foreach (var keyValuePair in dict)
            {
                if (keyValuePair.Key == "date")
                { externalDate = keyValuePair.Value.S; }
                if (keyValuePair.Key == "temp")
                { externalTemp = keyValuePair.Value.S; }
                if (keyValuePair.Key == "light")
                { externalLight = keyValuePair.Value.S; }
```

```
                    if (keyValuePair.Key == "humidity")
                    { externalHumidity = keyValuePair.Value.S; }
               }
               Debug.Log("Values for external_sensors are external date =" + externalDate +
" externalTemp=" + externalTemp + " externalLight= " + externalLight + " externalHumidity=
 " + externalHumidity);
          }
     }
}
```

D.5 camera_movement.cs

```
ï»¿using UnityEngine;
using System.Collections;

public class camera_movement : MonoBehaviour
{

    /*
     * Based on Windex's flycam script found here: http://forum.unity3d.com/threads/fly-cam-simple-cam-script.67042/
     * C# conversion created by Ellandar
     * Improved camera made by LookForward
     * Modifications created by Angryboy
     * 1) Have to hold right-click to rotate
     * 2) Made variables public for testing/designer purposes
     * 3) Y-axis now locked (as if space was always being held)
     * 4) Q/E keys are used to raise/lower the camera
     */

    public float mainSpeed = 100.0f; // regular speed
    public float shiftAdd = 250.0f; // multiplied by how long shift is held.  Basically running
    public float maxShift = 1000.0f; //Maximum speed when holding shift
    public float camSens = 0.25f; //How sensitive it with mouse
    // private Vector3 lastMouse = new Vector3(255, 255, 255); //kind of in the middle of the screen, rather than at the top (play)
     private float totalRun = 1.0f;

     private bool isRotating = false; // Angryboy: Can be called by other things (e.g. UI) to see if camera is rotating
     private float speedMultiplier; // Angryboy: Used by Y axis to match the velocity on X/Z axis

    public float mouseSensitivity = 5.0f;        // Mouse rotation sensitivity.
     private float rotationY = 0.0f;


    void Update()
    {

        // Angryboy: Hold right-mouse button to rotate
        if (Input.GetMouseButtonDown(1))
        {
            isRotating = true;
        }
        if (Input.GetMouseButtonUp(1))
        {
            isRotating = false;
        }
        if (isRotating)
        {
            // Made by LookForward
            // Angryboy: Replaced min/max Y with numbers, not sure why we had variables in the first place
```

```
            float  rotationX = transform.localEulerAngles.y + Input.GetAxis("Mouse X") * mouseSensitivity
;
            rotationY += Input.GetAxis("Mouse Y") * mouseSensitivity;
            rotationY = Mathf.Clamp(rotationY, -90, 90);
            transform.localEulerAngles = new Vector3(-rotationY, rotationX, 0.0f);
        }

        // Keyboard commands
//          float f = 0.0f;
        Vector3 p = GetBaseInput();
        if (Input.GetKey(KeyCode.LeftShift))
        {
            totalRun += Time.deltaTime;
            p = p * totalRun * shiftAdd;
            p.x = Mathf.Clamp(p.x, -maxShift, maxShift);
            p.y = Mathf.Clamp(p.y, -maxShift, maxShift);
            p.z = Mathf.Clamp(p.z, -maxShift, maxShift);
            // Angryboy: Use these to ensure that Y-plane is affected by the shift key as well
            speedMultiplier = totalRun * shiftAdd * Time.deltaTime;
            speedMultiplier = Mathf.Clamp(speedMultiplier, -maxShift, maxShift);
        }
        else
        {
            totalRun = Mathf.Clamp(totalRun * 0.5f, 1f, 1000f);
            p = p * mainSpeed;
            speedMultiplier = mainSpeed * Time.deltaTime; // Angryboy: More "correct" speed
        }

        p = p * Time.deltaTime;

        // Angryboy: Removed key-press requirement, now perma-locked to the Y plane
        Vector3 newPosition = transform.position; // If player wants to move on X and Z axis only
        transform.Translate(p);
        newPosition.x = transform.position.x;
        newPosition.z = transform.position.z;

        // Angryboy: Manipulate Y plane by using Q/E keys
        if (Input.GetKey(KeyCode.Q))
        {
            newPosition.y += -speedMultiplier;
        }
        if (Input.GetKey(KeyCode.E))
        {
            newPosition.y += speedMultiplier;
        }

        transform.position = newPosition;
    }

    // Angryboy: Can be called by other code to see if camera is rotating
    // Might be useful in UI to stop accidental clicks while turning?
    public bool amIRotating()
    {
        return isRotating;
```

```
    }

    private Vector3 GetBaseInput()
    { // returns the basic values, if it's 0 than it's not active.
        Vector3 p_Velocity = new Vector3();
        if (Input.GetKey(KeyCode.W))
        {
            p_Velocity += new Vector3(0, 0, 1);
        }
        if (Input.GetKey(KeyCode.S))
        {
            p_Velocity += new Vector3(0, 0, -1);
        }
        if (Input.GetKey(KeyCode.A))
        {
            p_Velocity += new Vector3(-1, 0, 0);
        }
        if (Input.GetKey(KeyCode.D))
        {
            p_Velocity += new Vector3(1, 0, 0);
        }
        return p_Velocity;
    }
}
```

D.6 CameraController.cs

```
ï»¿using UnityEngine;
using System.Collections;
using UnityEngine.Events;

public class CameraController : MonoBehaviour
{
    public Camera[] cameras;
    private int currentCameraIndex;

    // Use this for initialization
    void Start ()
    {
        currentCameraIndex = 0;

        //Turn all cameras off, except the first default one
        for (int i = 1; i < cameras.Length; i++)
        {
            cameras[i].gameObject.SetActive(false);
        }

        //If any cameras were added to the controller, enable the first one
        if (cameras.Length > 0)
        {
            cameras[0].gameObject.SetActive(true);
            Debug.Log("Camera with name: " + cameras[0].GetComponent<Camera>().name + ", is
now enabled");
        }
    }

    // Update is called once per frame
    void Update()
    {
        //If the c button is pressed, switch to the next camera
        //Set the camera at the current index to inactive, and set the next one in the array to active
        //When we reach the end of the camera array, move back to the beginning or the array.
        if (Input.GetKeyDown(KeyCode.C))
        {
            currentCameraIndex++;
            Debug.Log("C button has been pressed. Switching to the next camera");
            if (currentCameraIndex < cameras.Length)
            {
                cameras[currentCameraIndex - 1].gameObject.SetActive(false);
                cameras[currentCameraIndex].gameObject.SetActive(true);
                Debug.Log("Camera with name: " + cameras[currentCameraIndex].GetComponent<Camera
>().name + ", is now enabled");
            }
            else
            {
                cameras[currentCameraIndex - 1].gameObject.SetActive(false);
                currentCameraIndex = 0;
                cameras[currentCameraIndex].gameObject.SetActive(true);
```

```
                    Debug.Log("Camera with name: " + cameras[currentCameraIndex].GetComponent<Camera>().name + ", is now enabled");
                }
            }
        }

        public void OnClick()
        {
            currentCameraIndex++;
            Debug.Log("Button has been clicked Camera Changing");
            if (currentCameraIndex < cameras.Length)
            {
                cameras[currentCameraIndex - 1].gameObject.SetActive(false);
                cameras[currentCameraIndex].gameObject.SetActive(true);
                Debug.Log("Camera with name: " + cameras[currentCameraIndex].GetComponent<Camera>().name + ", is now enabled");
            }
            else
            {
                cameras[currentCameraIndex - 1].gameObject.SetActive(false);
                currentCameraIndex = 0;
                cameras[currentCameraIndex].gameObject.SetActive(true);
                Debug.Log("Camera with name: " + cameras[currentCameraIndex].GetComponent<Camera>().name + ", is now enabled");
            }
        }
}
```

D.7 mouseDrag.cs

```
ï»¿using UnityEngine;
using System.Collections;

public class mouseDrag : MonoBehaviour {

    public float distance = 0.0f;

void OnMouseDown()
    {
    Vector3 heading = this.transform.position - Camera.main.transform.position;
     distance = Vector3.Dot(heading, Camera.main.transform.forward);
    }

void OnMouseDrag()
    {
        Vector3 mousePosition = new Vector3(Input.mousePosition.x, Input.mousePosition.y, distance);
        Vector3 objPosition = Camera.main.ScreenToWorldPoint(mousePosition);
        transform.position = objPosition;
    }
}
```

D.8 quit.cs

```
ï»¿using UnityEngine;
using System.Collections;

public class quit : MonoBehaviour
{
    void LateUpdate()
    {
        if (Input.GetKey("escape"))
            Application.Quit();
    }

    public void OnClick()
    {
        Application.Quit();
    }
}
```

D.9 resetVessels.cs

```
using UnityEngine;
using System.Collections;

public class resetVessels : MonoBehaviour
{
    public GameObject vessel1;
    public GameObject vessel2;
    public GameObject vessel3;
    public GameObject vessel4;

    private Transform vessel1Transform, vessel2Transform, vessel3Transform, vessel4Transform;

    void Start ()
    {
        vessel1Transform = getVesselTransform(vessel1);
        vessel2Transform = getVesselTransform(vessel2);
        vessel3Transform = getVesselTransform(vessel3);
        vessel4Transform = getVesselTransform(vessel4);
    }

    void Update()
    {
        if (Input.GetKeyDown(KeyCode.Space))
            OnClick();
    }

    // Use this to get initial Transforms of Vessels
    private Transform getVesselTransform(GameObject Transforms)
    {
        Transform vesselTransform = new GameObject().transform;
        Vector3 position = Transforms.transform.position;
        Quaternion quaternion = Transforms.transform.rotation;
        vesselTransform.position = position;
        vesselTransform.rotation = quaternion;
        return vesselTransform;
    }

    // Reset Transforms of Vessels to original Transforms
    public void OnClick()
    {
        vessel1.transform.position = vessel1Transform.position;
        vessel1.transform.rotation = vessel1Transform.rotation;
//          Debug.Log("Setvessel1Position" + vessel1Transform.position);
//          Debug.Log("Setvessel1Rotation" + vessel1Transform.rotation);
        vessel1.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
        vessel1.GetComponent<Rigidbody>().angularVelocity = new Vector3(0, 0, 0);
        vessel2.transform.position = vessel2Transform.position;
        vessel2.transform.rotation = vessel2Transform.rotation;
        vessel2.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
        vessel2.GetComponent<Rigidbody>().angularVelocity = new Vector3(0, 0, 0);
        vessel3.transform.position = vessel3Transform.position;
        vessel3.transform.rotation = vessel3Transform.rotation;
```

```
        vessel3.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
        vessel3.GetComponent<Rigidbody>().angularVelocity = new Vector3(0, 0, 0);
        vessel4.transform.position = vessel4Transform.position;
        vessel4.transform.rotation = vessel4Transform.rotation;
        vessel4.GetComponent<Rigidbody>().velocity = new Vector3(0, 0, 0);
        vessel4.GetComponent<Rigidbody>().angularVelocity = new Vector3(0, 0, 0);
    }
}
```

D.10 snap2.js

```
//#pragma strict ;

var partnerName = "Hub";
var closeVPDist = 0.05;
var moveSpeed = 40.0;
var rotateSpeed = 90.0;

var closeColor = Color (1,0,0) ;
var partnerColor = Color (1,0,0) ;

private var dist = Mathf. Infinity ;
private var normalColor : Color;
private var partnerGO : GameObject;

function Start () {
    normalColor = GetComponent.<Renderer>().material. color ;
    partnerGO = GameObject.Find(partnerName);
    partnerGO.GetComponent(MeshRenderer).enabled = false;
}

function OnMouseDrag() {
    var partnerPos = Camera.main.WorldToViewportPoint(partnerGO.transform. position ) ;
    var myPos = Camera.main.WorldToViewportPoint(transform.position ) ;
    dist = Vector2 . Distance ( partnerPos , myPos);
    GetComponent.<Renderer>().material. color = ( dist < closeVPDist) ? closeColor : normalColor;
    partnerGO = GameObject.Find(partnerName);
    partnerGO.GetComponent.<Renderer>().material. color = partnerColor ;
    partnerGO.GetComponent(MeshRenderer).enabled = true;
}

function OnMouseUp() {
    if ( dist < closeVPDist) {
        transform . parent = partnerGO.transform ;
        GetComponent.<Renderer>().material. color = normalColor;
        InstallPart () ;
        partnerGO = GameObject.Find(partnerName);
        partnerGO.GetComponent(MeshRenderer).enabled = false;
    }
}

function InstallPart () {
    while (transform . localPosition != Vector3 . zero || transform . localRotation != Quaternion. identity ) {
        transform . localPosition = Vector3 . MoveTowards(transform. localPosition , Vector3 . zero , Time.
deltaTime * moveSpeed);
        transform . localRotation = Quaternion . RotateTowards(transform . localRotation , Quaternion. identity ,
Time.deltaTime * rotateSpeed ) ;
        yield ;
    }
}
//      GetComponent(MeshRenderer).enabled = true;
```

D.11 PlayerController3D.cs

```
ï»¿using UnityEngine;
using System.Collections;
using UnityEngine.Networking;

public class PlayerController3D : NetworkBehaviour
{
    [SerializeField]
    private float mainSpeed = 10.0f; // regular speed
    private float shiftAdd = 100.0f; // multiplied by how long shift is held. Basically running
    private float maxShift = 1000.0f; // Maximum speed when holdin gshift
    // private float camSens = 0.25f; // How sensitive it with mouse
    // private Vector3 lastMouse = new Vector3(255, 255, 255); // kind of in the middle of the screen, rather than at the top (play)
    private float totalRun = 1.0f;

    private bool isRotating = false; // Angryboy: Can be called by other things (e.g. UI) to see if camera is rotating
    private float speedMultiplier; // Angryboy: Used by Y axis to match the velocity on X/Z axis

    private float mouseSensitivity = 5.0f;         // Mouse rotation sensitivity.
    private float rotationY = 0.0f;
    public GameObject Player;

    void Update()
    {
        if (!isLocalPlayer)
        {
            return;
        }
        // Angryboy: Hold right-mouse button to rotate
        if (Input.GetMouseButtonDown(1))
        {
            isRotating = true;
        }
        if (Input.GetMouseButtonUp(1))
        {
            isRotating = false;
        }
        if (isRotating)
        {
            // Made by LookForward
            // Angryboy: Replaced min/max Y with numbers, not sure why we had variables in the first place
            float rotationX = transform.localEulerAngles.y + Input.GetAxis("Mouse X") * mouseSensitivity;
            rotationY += Input.GetAxis("Mouse Y") * mouseSensitivity;
            rotationY = Mathf.Clamp(rotationY, -90, 90);
            transform.localEulerAngles = new Vector3(-rotationY, rotationX, 0.0f);
        }

        //Keyboard commands
//          float f = 0.0f;
```

```
        Vector3 p = GetBaseInput();
        if (Input.GetKey(KeyCode.LeftShift))
        {
            totalRun += Time.deltaTime;
            p = p * totalRun * shiftAdd;
            p.x = Mathf.Clamp(p.x, -maxShift, maxShift);
            p.y = Mathf.Clamp(p.y, -maxShift, maxShift);
            p.z = Mathf.Clamp(p.z, -maxShift, maxShift);
            // Angryboy: Use these to ensure that Y-plane is affected by the shift key as well
            speedMultiplier = totalRun * shiftAdd * Time.deltaTime;
            speedMultiplier = Mathf.Clamp(speedMultiplier, -maxShift, maxShift);
        }
        else
        {
            totalRun = Mathf.Clamp(totalRun * 0.5f, 1f, 1000f);
            p = p * mainSpeed;
            speedMultiplier = mainSpeed * Time.deltaTime; // Angryboy: More "correct" speed
        }

        p = p * Time.deltaTime;

        // Angryboy: Removed key-press requirement, now perma-locked to the Y plane
        Vector3 newPosition = transform.position; // If player wants to move on X and Z axis only
        transform.Translate(p);
        newPosition.x = transform.position.x;
        newPosition.z = transform.position.z;

        // Angryboy: Manipulate Y plane by using Q/E keys
        if (Input.GetKey(KeyCode.Q))
        {
            newPosition.y += -speedMultiplier;
        }
        if (Input.GetKey(KeyCode.E))
        {
            newPosition.y += speedMultiplier;
        }

        transform.position = newPosition;
    }

    // Angryboy: Can be called by other code to see if camera is rotating
    // Might be useful in UI to stop accidental clicks while turning?
    public bool amIRotating()
    {
        return isRotating;
    }

    private Vector3 GetBaseInput()
    { // returns the basic values, if it's 0 than it's not active.
        Vector3 p_Velocity = new Vector3();
        if (Input.GetKey(KeyCode.W))
        {
            p_Velocity += new Vector3(0, 0, 1);
        }
```

```
        if (Input.GetKey(KeyCode.S))
        {
            p_Velocity += new Vector3(0, 0, -1);
        }
        if (Input.GetKey(KeyCode.A))
        {
            p_Velocity += new Vector3(-1, 0, 0);
        }
        if (Input.GetKey(KeyCode.D))
        {
            p_Velocity += new Vector3(1, 0, 0);
        }
        return p_Velocity;
    }
    public override void OnStartLocalPlayer()
    {
        Player = this.gameObject;
    }
}
```

D.12 PlayerSetup.cs

```
ï»¿using UnityEngine;
using UnityEngine.Networking;
using UnityEngine.VR;
using Dissonance.VAD;

public class PlayerSetup : NetworkBehaviour {

    public Camera playerCamera;
    [ SerializeField ] Behaviour[] componentsToDisable;
    [ SerializeField ] Behaviour[] disableIfNotVR;
    [ SerializeField ] GameObject VRTK;
    Camera sceneCamera;
    [ SerializeField ] GameObject[] componentsToSpawn;
    public GameObject PlayerCapsule;
    bool cullingMaskOnOff;

    private void Start ()
    {
        if (! isLocalPlayer )
        {
            for ( int i = 0; i < componentsToDisable.Length; i++)
            {
                componentsToDisable[i].enabled = false ;
            }
        }
        else
        {
            sceneCamera = Camera.main;
            if (sceneCamera != null )
            {
                sceneCamera.gameObject.SetActive( false ) ;
            }
        }

        if ( isLocalPlayer && VRDevice.isPresent) //Only enable VRTK if VRDevice is preset , and only on
the local player .
        {
            VRTK.SetActive(true);
        }

        if ( isLocalPlayer && !VRDevice.isPresent) // only disable if VRDevice is not present , and only on
the local player .
        {
            for ( int i = 0; i < disableIfNotVR.Length; i++)
            {
                disableIfNotVR[i ]. enabled = false ;
            }
        }
    }

    private void OnDisable()
    {
```

```
        if (sceneCamera != null)
        {
            sceneCamera.gameObject.SetActive(true);
        }
    }

    private void Awake()
    {
        for (int i = 0; i < componentsToSpawn.Length; i++)
        {
            Vector3 spawnPosition = new Vector3(0, 0, 0);
//          Quaternion spawnRotation = Quaternion.Euler(0, 0, 0);
            GameObject spawnObject = (GameObject)Instantiate(componentsToSpawn[i], spawnPosition,
transform.rotation);
            spawnObject.SetActive(true);
            Debug.Log("spawnObject is " + spawnObject.name);
            NetworkServer.Spawn(spawnObject);
        }
    }

    public void Update()
    {
        if (Input.GetKeyDown(KeyCode.P))
        {
            cullingMaskOnOff = !cullingMaskOnOff;

            if (cullingMaskOnOff == true)
            {
                playerCamera.cullingMask = (1 << LayerMask.NameToLayer("ViewCamCull")) | (1 <<
LayerMask.NameToLayer("PickUpLayer"));
            }
            else
            {
                playerCamera.cullingMask = -1;
            }
        }
    }

    public override void OnStartLocalPlayer()
    {
        PlayerCapsule.GetComponent<MeshRenderer>().material.color = Color.blue;
    }
}
```

D.13 WorkInstructionsNetworked.cs

```
ï»¿using UnityEngine;
using UnityEngine.Events;
using UnityEngine.Networking;

public class WorkInstructionsNetworked : NetworkBehaviour
{
    private GameObject[] assemblyGuides; // create and array that will hold all the assemblies in the work instructions
    private int assemblyGuidesIndex; // this is the index counter for what work instruction we have reached
    [SyncVar]
    private GameObject assemblyObjOff;
    [SyncVar]
    private GameObject assemblyObjOn;
    [SyncVar]
    private NetworkIdentity assemblyObjOffNetID;
    [SyncVar]
    private NetworkIdentity assemblyObjOnNetID;
    bool WorkInstructionAllOnOff;
//    private GameObject workInstructionsGo;

    // Use this for initialization
    void Awake()
    {
        assemblyGuidesIndex = 0;
//        workInstructionsGo = GameObject.FindWithTag("GUI2");
        assemblyGuides = GameObject.FindWithTag("WorkInstructions").GetComponent<WorkInstructions>().assemblyGuidesArray;
        Debug.Log("This is the awake function on the PLAYER the first member of the array is " + assemblyGuides[0] + "The second is " + assemblyGuides[1]);
    }

    void Start ()
    {
        GameObject.FindWithTag("GUI2").GetComponent<WorkInstructionsGUI>().playerWorkInstructions = this; // identify this as the workinstructions for GUI calls
    }

    void Update()
    {
        if ( isLocalPlayer )
        {
            CheckWorkInstruction();
        }
    }

    public void moveWorkInstructionForward()
    {
        assemblyGuidesIndex++;
        Debug.Log("I button has been pressed. Switching to the next Work Instruction" + assemblyGuidesIndex);
        if (assemblyGuidesIndex < assemblyGuides.Length)
```

```
        {
            assemblyObjOff = assemblyGuides[assemblyGuidesIndex − 1];
            assemblyObjOn = assemblyGuides[assemblyGuidesIndex];
            CmdWorkInstructionProgress(assemblyObjOff, assemblyObjOn);
        }
        else
        {
            assemblyObjOff = assemblyGuides[assemblyGuidesIndex − 1];
            assemblyGuidesIndex = 0;
            assemblyObjOn = assemblyGuides[assemblyGuidesIndex];
            CmdWorkInstructionProgress(assemblyObjOff, assemblyObjOn);
        }
    }

    public void moveWorkInstructionBackwards()
    {
        assemblyGuidesIndex−−;
        Debug.Log("I button has been pressed. Switching to the next Work Instruction" + assemblyGuidesIndex);
        if (assemblyGuidesIndex <= 0)
        {
            assemblyObjOff = assemblyGuides[assemblyGuidesIndex + 1];
            assemblyGuidesIndex = assemblyGuides.Length;
            assemblyObjOn = assemblyGuides[assemblyGuidesIndex];
            CmdWorkInstructionProgress(assemblyObjOff, assemblyObjOn);
        }
        else
        {
            assemblyObjOff = assemblyGuides[assemblyGuidesIndex + 1];
            assemblyObjOn = assemblyGuides[assemblyGuidesIndex];
            CmdWorkInstructionProgress(assemblyObjOff, assemblyObjOn);
        }
    }

    public void workInstructionsOnOff ()
    {
        WorkInstructionAllOnOff = !WorkInstructionAllOnOff;

        if (WorkInstructionAllOnOff)
        {
            CmdTurnAllOff(assemblyGuides);
        }
        else
        {
            CmdTurnAllOn(assemblyGuides);
        }
    }

    void CheckWorkInstruction()
    {
        if ( isLocalPlayer && Input.GetKeyDown(KeyCode.I)) // isLcoalPlayer && I for instruction move forward
        {
            moveWorkInstructionForward();
```

```
        }

        if ( isLocalPlayer  && Input.GetKeyDown(KeyCode.U))
        {
            workInstructionsOnOff () ;
        }
    }

    [ClientRpc]
    void  RpcInstructionProgress (GameObject assemblyObjOff, GameObject assemblyObjOn)            // this  is
 where we really  turn  the  assemblies  on and  off
    {
        assemblyObjOff.SetActive( false ) ;
        assemblyObjOn.SetActive(true) ;
    }

    [Command]
    void  CmdWorkInstructionProgress(GameObject assemblyObjOff, GameObject assemblyObjOn)
    {
        assemblyObjOffNetID = assemblyObjOff.GetComponent<NetworkIdentity>();          //  get  the  object ' s
network ID
        assemblyObjOnNetID = assemblyObjOn.GetComponent<NetworkIdentity>();         //  get  the  object ' s
network ID
        assemblyObjOffNetID.AssignClientAuthority( connectionToClient ) ;       //  assign  authority  to  the
 player  who is  changing  the  state
        assemblyObjOnNetID.AssignClientAuthority(connectionToClient ) ;       //  assign  authority  to  the
 player  who is  changing  the  state

         RpcInstructionProgress (assemblyObjOff, assemblyObjOn);                                  //  use
a  Client  RPC function  to  change  assemblies  on  all   clients

        assemblyObjOffNetID.RemoveClientAuthority(connectionToClient);       //  remove the  authority  from
 the  player  who changed the  state
        assemblyObjOnNetID.RemoveClientAuthority(connectionToClient);       //  remove the  authority  from the
  player  who changed the  state
    }

    [ ClientRpcAttribute ]
    void  RpcTurnAllOff(GameObject[] assemblyGuides)
    {
        for  ( int  i  =  0;  i  < assemblyGuides.Length;  i++)
        {
            assemblyGuides[i ]. gameObject.SetActive( false ) ;
        }
    }

    [ ClientRpcAttribute ]
    void  RpcTurnAllOn(GameObject[] assemblyGuides)
    {
        for  ( int  i  =  0;  i  < assemblyGuides.Length;  i++)
        {
            assemblyGuides[i ]. gameObject.SetActive( true ) ;
        }
    }
```

```

    [Command]
    void CmdTurnAllOff(GameObject[] assemblyGuides)
    {
        RpcTurnAllOff(assemblyGuides);
    }

    [Command]
    void CmdTurnAllOn(GameObject[] assemblyGuides)
    {
        RpcTurnAllOn(assemblyGuides);
    }
}
```

D.14 GUI_onOff.cs

```
ï»¿using UnityEngine;
using UnityEngine.Events;
using UnityEngine.Networking;


public class GUI_OnOff : NetworkBehaviour {

    private GameObject GUI_element_1;
    private GameObject GUI_element_2;

    bool GUI_AllOnOff;

    void Awake()
    {
        GUI_element_1 = GameObject.FindGameObjectWithTag("GUI1"); //find the local player
        Debug.Log("GUI_element_1 " + GUI_element_1);
        GUI_element_2 = GameObject.FindGameObjectWithTag("GUI2"); //find the local player
        Debug.Log("GUI_element_2 " + GUI_element_2);
    }

    void Update()
    {
        if ( isLocalPlayer )
        {
            GUIOnOff();
        }
    }

    void GUIOnOff()
    {
        if ( isLocalPlayer && Input.GetKeyDown(KeyCode.G))
        {
            GUI_AllOnOff = !GUI_AllOnOff;
            Debug.Log("keypress G");
            if (GUI_AllOnOff)
            {
                GUI_element_1.GetComponent<Canvas>().enabled = false; // turn off the canvas component not the game object which has other scripts on it
                GUI_element_2.GetComponent<Canvas>().enabled = false;
                Debug.Log("GUI_AllOnOff " + GUI_AllOnOff);
            }
            else
            {
                GUI_element_1.GetComponent<Canvas>().enabled = true;
                GUI_element_2.GetComponent<Canvas>().enabled = true;
            }

        }
    }

}
```

D.15 NetworkGrabManager.cs

```
ï»¿using UnityEngine;
using UnityEngine.Events;
using UnityEngine.Networking;
using VRTK;

public class NetworkGrabManager : NetworkBehaviour{

    public GameObject leftController ;
    public GameObject rightController ;
    [SyncVar] private NetworkIdentity objNetId;

    void Awake()
    {
            Debug.Log("[NETWORK GRAB MANAGER] This is the NetWorkGrabManager OnAwake");

//          VRTK_InteractGrab objLeft = leftController .GetComponent<VRTK_InteractGrab>();
//          VRTK_InteractGrab objRight = rightController .GetComponent<VRTK_InteractGrab>();

//          objLeft . ControllerGrabInteractableObject += Obj_ControllerGrabInteractableObject ;
//          objRight . ControllerGrabInteractableObject += Obj_ControllerGrabInteractableObject ;

//          objLeft . ControllerUngrabInteractableObject += Obj_ControllerUngrabInteractableObject ;
//          objRight . ControllerUngrabInteractableObject += Obj_ControllerUngrabInteractableObject ;

            // attempt the same process but with touch and untouch also keep separate hands separate
            VRTK_InteractTouch objLeftTouch = leftController .GetComponent<VRTK_InteractTouch>();
            VRTK_InteractTouch objRightTouch = rightController .GetComponent<VRTK_InteractTouch>();

            objLeftTouch. ControllerTouchInteractableObject += ObjLeftTouch_ControllerTouchInteractableObject ;
            objRightTouch. ControllerTouchInteractableObject += ObjRightTouch_ControllerTouchInteractableObject ;

            objLeftTouch. ControllerUntouchInteractableObject += ObjLeftTouch_ControllerUntouchInteractableObject ;
            objRightTouch. ControllerUntouchInteractableObject += ObjRightTouch_ControllerUntouchInteractableObject ;
    }

    [Command]
    private void CmdRemoveClientAuthority(GameObject unGrabbedObject)
    {
        objNetId = unGrabbedObject.GetComponent<NetworkIdentity>();
        objNetId.RemoveClientAuthority(connectionToClient);
    }

    [Command]
    private void CmdAssignClientAuthority(GameObject grabbedObject)
    {
        objNetId = grabbedObject.GetComponent<NetworkIdentity>();
        objNetId. AssignClientAuthority ( connectionToClient );
```

```
    }

    private void ObjRightTouch_ControllerTouchInteractableObject ( object sender, ObjectInteractEventArgs e
)
    {
        GameObject grabbedObject = e. target ;
        // if object is grabbable assign authority
        if (grabbedObject.GetComponent<VRTK_InteractableObject>() == true)
        {
            Debug.Log(" [NETWORK GRAB MANAGER] Touched Object is " + grabbedObject.name);
            CmdAssignClientAuthority(grabbedObject);
        }
    }

    private void ObjLeftTouch_ControllerTouchInteractableObject ( object sender, ObjectInteractEventArgs e)
    {
        GameObject grabbedObject = e. target ;
        // if object is grabbable assign authority
        if (grabbedObject.GetComponent<VRTK_InteractableObject>()== true)
        {
            Debug.Log(" [NETWORK GRAB MANAGER] Touched Object is " + grabbedObject.name);
            CmdAssignClientAuthority(grabbedObject);
        }
    }

    private void ObjRightTouch_ControllerUntouchInteractableObject ( object sender, ObjectInteractEventArgs
 e)
    {
        GameObject unGrabbedObject = e.target ;
        // if object is grabbable remove authority
        if (unGrabbedObject.GetComponent<VRTK_InteractableObject>() == true)
        {
            Debug.Log(" [NETWORK GRAB MANAGER] unTouched Object is " + unGrabbedObject.
name);
            CmdRemoveClientAuthority(unGrabbedObject);
        }
    }

    private void ObjLeftTouch_ControllerUntouchInteractableObject ( object sender, ObjectInteractEventArgs
e)
    {
        GameObject unGrabbedObject = e.target ;
        // if object is grabbable remove authority
        if (unGrabbedObject.GetComponent<VRTK_InteractableObject>() == true)
        {
            Debug.Log(" [NETWORK GRAB MANAGER] unTouched Object is " + unGrabbedObject.
name);
            CmdRemoveClientAuthority(unGrabbedObject);
        }
    }

/*    private void Obj_ControllerGrabInteractableObject ( object sender, ObjectInteractEventArgs e)
    {
        GameObject grabbedObject = e. target ;
```

```
            Debug.Log("[NETWORK GRAB MANAGER] Grabbed Object is " + grabbedObject.name);
            CmdAssignClientAuthority(grabbedObject);
        }

        private void Obj_ControllerUngrabInteractableObject(object sender, ObjectInteractEventArgs e)
        {
            GameObject unGrabbedObject = e.target;
            Debug.Log("[NETWORK GRAB MANAGER] unGrabbed Object is " + unGrabbedObject.name);
            CmdRemoveClientAuthority(unGrabbedObject);
        }*/
}
```

D.16 PickUpScript.cs

```
ï»¿using UnityEngine;
using System.Collections;
using UnityEngine.Networking;

public class PickUpScript : NetworkBehaviour
{
    [SerializeField] private float distance = 0.0f;
    [SerializeField] private Camera playerCamera;
    [SerializeField] private GameObject playerCapsule;
    private RaycastHit hit;
    [SyncVar] private GameObject objectID;
    [SyncVar] private NetworkIdentity objNetID;
    [SyncVar] private Color objectColor;
    [SyncVar] private GameObject obj;

    private int jumpToLayerMask = 1 << 10; //jumpToUpLayer is 10
    private int pickUpLayerMask = 1 << 9; //pickUpLayer is 9

// trying to avoid using update function
// on mouse down assign authority
// on mouse drag move the shizzle

    //      void Start ()
    public override void OnStartLocalPlayer () // re-initiate for each local player start
    {
        if (!isLocalPlayer)
        {
            return;
        }
        GameObject.Find("Side Menu Canvas").GetComponent<WorkInstructionsGUI>().playerJumpTo =
this; // indentify this as the JumptInstructions for GUI calls
    }

    void Update()
    {
        if (isLocalPlayer && Input.GetMouseButton(0))
        {
            CheckIfMoving();
            CheckWhichCamera();
        }
    }

    [Command]
    void CmdRemoveClientAuthority()
    {
        objNetID = objectID.GetComponent<NetworkIdentity>();    // get the object's network ID
        objNetID.RemoveClientAuthority(connectionToClient);
    }

    [ClientRpc]
    void RpcMoveTo(GameObject obj, Color col)
    {
```

```
        obj.GetComponent<Renderer>().material.color = col;        // this is the line that actually makes the change in color happen
        Debug.Log("This is obj" + obj.name);
        this.transform.position = obj.GetComponent<cameraMoveToLocation>().moveToPosition;
        this.transform.rotation = obj.GetComponent<cameraMoveToLocation>().QMoveToRotation;
    }

    [Command]
    void CmdPickUp(GameObject objectID)
    {
        objNetID = objectID.GetComponent<NetworkIdentity>();        // get the object's network ID
        objNetID.AssignClientAuthority(connectionToClient);      // assign authority to the player who is changing the color

        Vector3 mousePosition = new Vector3(Input.mousePosition.x, Input.mousePosition.y, distance);
//          Vector3 mousePosition = new Vector3(Input.mousePosition.x, Input.mousePosition.y - (Input.mousePosition.y/5), distance); //mouse Y position adjusted due to issues with running two instances on one device
        Vector3 objPosition = playerCamera.ScreenToWorldPoint(mousePosition);
        Quaternion objRotation = playerCamera.transform.rotation;

        objectID.transform.position = objPosition;
        objectID.transform.rotation = objRotation;

 //         this.GetComponent<Renderer>().material.color = Color.clear; // finally change color to transparent

       objNetID.RemoveClientAuthority(connectionToClient);      // remove the authority from the player who changed the color
    }

    [Command]
    void CmdMoveTo(GameObject obj, Color col)
    {
        objNetID = obj.GetComponent<NetworkIdentity>();         // get the object's network ID
        objNetID.AssignClientAuthority(connectionToClient);      // assign authority to the player who is changing the color
        RpcMoveTo(obj, col);                                     // use a Client RPC function to "paint" the object on all clients
        playerCapsule.GetComponent<Renderer>().material.color = Color.clear; //change player color to transparent
        objNetID.RemoveClientAuthority(connectionToClient);      // remove the authority from the player who changed the color
    }

    [Command]
    void CmdClearPlayerColor()
    {
        playerCapsule.GetComponent<Renderer>().material.color = Color.clear; //change player color to transparent
    }

    public void MoveToProjectorOne()
    {
```

```
            obj = GameObject.FindWithTag("Projector1");
            objectColor = new Color(Random.value, Random.value, Random.value, Random.value);    // choose new random color
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors -- not needed for Android GUI
            CmdMoveTo(obj, objectColor);
            CmdClearPlayerColor();
      }

      public void MoveToProjectorTwo()
      {
            obj = GameObject.FindWithTag("Projector2");
            objectColor = new Color(Random.value, Random.value, Random.value, Random.value);    // choose new random color
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors -- -- not needed for Android GUI
            CmdMoveTo(obj, objectColor);
            CmdClearPlayerColor();
      }

      public void MoveToProjectorThree()
      {
            obj = GameObject.FindWithTag("Projector3");
            objectColor = new Color(Random.value, Random.value, Random.value, Random.value);    // choose new random color
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors -- -- not needed for Android GUI
            CmdMoveTo(obj, objectColor);
            CmdClearPlayerColor();
      }

      public void MoveToExtProjectorOne()
      {
            obj = GameObject.Find("ExtProjector1");
            objectColor = new Color(Random.value, Random.value, Random.value, Random.value);    // choose new random color
            Camera.main.fieldOfView = 60; //change the FoV to meet that of normal game views
            CmdMoveTo(obj, objectColor);
            CmdClearPlayerColor();
      }

      public void MoveToExtProjectorTwo()
      {
            obj = GameObject.Find("ExtProjector2");
            objectColor = new Color(Random.value, Random.value, Random.value, Random.value);    // choose new random color
            Camera.main.fieldOfView = 60; //change the FoV to meet that of normal game views
            CmdMoveTo(obj, objectColor);
            CmdClearPlayerColor();
      }

      void CheckWhichCamera()
      {
            if ( isLocalPlayer && Input.GetKeyDown(KeyCode.Alpha1))
```

```
        {
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors
            MoveToProjectorOne();
        }
        if (isLocalPlayer && Input.GetKeyDown(KeyCode.Alpha2))
        {
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors
            MoveToProjectorTwo();
        }
        if (isLocalPlayer && Input.GetKeyDown(KeyCode.Alpha3))
        {
            Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors
            MoveToProjectorThree();
        }
    }

    void CheckIfMoving()
    {
        Ray ray = playerCamera.ScreenPointToRay(Input.mousePosition);

            if (Physics.Raycast(ray.origin, ray.direction, out hit, Mathf.Infinity, pickUpLayerMask))
            {
                objectID = GameObject.Find(hit.transform.name);                                   // this gets the object that is hit
//              Debug.Log("Object Hit is called " + objectID.name);
                CmdPickUp(objectID);
            }

        if (Input.GetKey(KeyCode.O))
        {
            if (Physics.Raycast(ray.origin, ray.direction, out hit, Mathf.Infinity, jumpToLayerMask))
            {
                Debug.DrawRay(transform.position, transform.TransformDirection(Vector3.forward) * hit.distance, Color.yellow);
                Debug.Log("Did Hit");
                obj = GameObject.Find(hit.transform.name);                                   // this gets the object that is hit
                Debug.Log(hit.transform.name);
                objectColor = new Color(Random.value, Random.value, Random.value, Random.value);     // I select the color here before doing anything else
                Camera.main.fieldOfView = 30; //change the FoV to meet that of the projectors
                CmdMoveTo(obj, objectColor);
            }
        }
    }
}
```

D.17 DissonanceVoIP_OnOff.cs

```
ï»¿using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using Dissonance;

public class DissonanceVoIP_OnOff : MonoBehaviour {

    bool VoiceBroadCastOnOff;

    public void VoiceBroadCastToggle()
    {
        VoiceBroadCastOnOff = !VoiceBroadCastOnOff;

        if (VoiceBroadCastOnOff)
        {
            DisableVoiceBroadCast();
        }
        else
        {
            EnableVoiceBroadCast();
        }
    }

    void EnableVoiceBroadCast()
    {
        GameObject.FindWithTag("Dissonance").GetComponent<VoiceBroadcastTrigger>().StartSpeaking(); //
find the dissonance gameobject and run the speaking function from the voice broadcast trigger script
    }

    void DisableVoiceBroadCast()
    {
        GameObject.FindWithTag("Dissonance").GetComponent<VoiceBroadcastTrigger>().StopSpeaking(); //
find the dissonance gameobject and run the speaking function from the voice broadcast trigger script
    }
}
```

Publications

Submissions to international conferences

- Christopher Freeman, Dr Rab Scott, **Stephen Reddish**: 'Discrete Event Simulation in Immersive Virtual Reality'. *Proceedings of the 4th Joint Virtual Reality Conference ICAT-EGVE-EuroVR*. JVRC, 2012.

Patents

- **Stephen Reddish**, Christopher Freeman, Michael Lewis: 'Apparatus, methods, computer programs, and non-transitory computer readable storage mediums for enabling remote control of one or more devices'. *UK: GB2549264A US: US20170293275A1 China: CN107422686A*

Acknowledgements

My grateful thanks are due to Dr Rab Scott, Mr Chris Freeman and Dr Martin Goodfellow for their professional advice and for giving me the opportunity to write this EngD thesis at the University of Sheffield & at Rolls-Royce.

I thank in particular the Nuclear AMRC team who readily provided information at any time. The advice and help provided by the Digital team at the AMRC Factory of the Future cannot be understated. In addition, the Nuclear EngD team at the University of Manchester kept me on track through their Business and Professional Development Modules.

Finally I would like to thank my partner and family, whose support has carried me through all the darker moments.

Without them all, I would not have been able to complete such an undertaking.