# An architecture for the integration of human workers into an Industry 4.0 manufacturing environment

by
Dale Eric Sparrow

*Dissertation presented for the degree of Doctor of Philosophy in the Faculty of Engineering at Stellenbosch University*

Supervisor: Dr Karel Kruger
Co-supervisor: Prof Anton Basson

March 2021

Departement Meganiese en Megatroniese Ingenieurswese
Department of Mechanical and Mechatronic Engineering

# Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: March 2021

# Abstract


## An architecture for the integration of human workers into an Industry 4.0 manufacturing environment



D.E. Sparrow

*Department of Mechanical and Mechatronic Engineering*
*Stellenbosch University*
*Private Bag X1, 7602 Matieland, South Africa*


Dissertation: Ph.D. (Mechatronic Engineering)
March 2021



With the rise of Industry 4.0 and the development in technologies that contribute to this revolution in manufacturing, research has focused mainly on the machines and automated digital systems contributing to the manufacturing environment. Humans are still critical to manufacturing; offering unmatched ingenuity, robustness, and flexibility despite their apparent disadvantages in strength or precision. Many successful manufacturing firms still include humans in their manufacturing processes for these reasons, and it is critical that the integration of humans in an I4.0 manufacturing environment is given research attention.

This dissertation first explores the requirements for the integration of human workers into an I4.0 environment. It was determined that the largest problem with human integration exists with data related to the human being digitised, managed, and communicated with other entities in processes that are identified as Administrative Logistics. It is identified that an administration shell similar to the RAMI4.0 administration shell concept is required to manage these Administrative Logistics on behalf of the human, and that a holonic systems approach is beneficial. The dissertation then proposes the concept of a Human Resource Holon Administration shell (HRH-AS).

An architecture to implement such an administration shell is then developed, here named the BASE architecture. This administration shell facilitates the interfacing, data processing, and connectivity to other I4.0 components on behalf of the human, to aid in their integration to the digital factory environment around them.

The BASE architecture addresses three identified responsibilities of such an administration shell, namely interfacing, digital data management, and delegation to other I4.0 components. BASE stands for Biography, Attributes, Schedule, and Execution, and is a time-based separation of concerns for key augmentations provided to the human worker. The BASE architecture separates value-adding and

decision-making plugin components, which are specific to an application, from the core components, which are generic to any application.

The BASE architecture led to the development of the 3SAL activity structure to facilitate the communication and management of industrial activities in a digital environment.

With the help of an industry partner, two case studies were developed to evaluate an implementation of the BASE architecture. The company is an aerospace composites manufacturer and was chosen for the labour-intensive requirements of the composites industry. The case studies aimed to evaluate the architecture against the three identified administration shell responsibilities and determine if the human workers are elevated to resource holon status. The first case study aimed to show how BASE facilitates interfacing with humans in an I4.0 environment and also acted as a technology demonstrator for the second case study. The second case study evaluated the effect BASE had on the Administrative Logistics involved in the business processes workers were involved in. Together these case studies fully evaluate BASE's ability to facilitate the integration of humans into an I4.0 manufacturing environment through identified responsibilities of the administration shell.

The evaluation found that the BASE HRH-AS improves the effectiveness of Administrative Logistics of business processes the human workers were involved with, as well as opened new opportunities for decision making on the shop floor previously not possible. Value-adding, by means of the plug-in components of a BASE administration shell, has also been proven by the ability to do automated schedule management, automatic calculation of standard work and improved traceability using the 3SAL activity structure.

# Uittreksel

## ‘n Argitektuur vir die integrasie van menslike werkers in ‘n Industrie 4.0 vervaardigingsomgewing

D.E. Sparrow

*Departement van Meganiese en Megatroniese Ingenieurswese*
*Universiteit Stellenbosch*
*Privaatsak X1, 7602 Matieland, Suid-Afrika*
Verhandeling: Ph.D. (Megatroniese Ingenieurswese)
Maart 2021

Met die opkoms van Industrie 4.0 (I4.0) en die ontwikkeling in tegnologieë wat bydra tot hierdie revolusie in vervaardiging, het navorsing grotendeels gefokus op die masjiene en digitale stelsels wat tot die vervaardigingsomgewing bydra. Menslike werkers is steeds van kritieke belang vir vervaardiging, en bied ongeëwenaarde vindingrykheid, robuustheid en buigsaamheid, ondanks hul oënskynlike nadele in sterkte of noukeurigheid. Baie suksesvolle vervaardigingsondernemings sluit nogsteeds mense in hul vervaardigingsprosesse in om hierdie redes. Dit is dus noodsaaklik dat die integrasie van mense met 'n I4.0 vervaardigingsomgewing navorsingsaandag geniet.

Hierdie verhandeling ondersoek eerstens die vereistes vir die integrasie van menslike werkers in 'n I4.0 omgewing. Daar word eerstens geïdentifiseer dat die grootste probleem met die integrasie van mense bestaan uit oneffektiewe prosesse wat data en informasie tussen mense en ander komponente in die vervaardigingsprosesse komminikeer, stoor en bestuur. Hierdie prosesse word na verwys as Administratiewe Logistieke. Daar word ook geïdentifiseer dat 'n administrasie-dop soortgelyk aan die RAMI4.0 administrasie-dop konsep benodig word om Administratiewe Logistieke te bestuur namens die mens, en dat 'n holoniese stelselbenadering voordelig is. Die verhandeling stel dan die konsep van 'n *Human Resource Holon Administration Shell* (HRH-AS) voor.

Daarna word 'n argitektuur ontwikkel om so 'n administrasie-dop te implementeer, hier genoem die BASE argitektuur. Hierdie administrasie-dop vergemaklik die inskakeling, dataverwerking en konnektiwiteit met ander I4.0 komponente namens die mens. Die integrasie van die mens in die digitale fabrieksomgewing is dus bevoordeel.

Die BASE argitektuur spreek drie geïdentifiseerde verantwoordelikhede van so 'n administrasie-dop aan, naamlik: koppelling met die mens, digitale databestuur, en delegering na ander I4.0 komponente namens die mens. BASE staan vir *Biography,*

*Attributes, Schedule* en *Execution*, en is 'n tydgebaseerde skeiding van belange vir aanvullings wat aan die menslike werker verskaf word. Die BASE argitektuur skei waardetoevoeging- en besluitneming-inpropkomponente, wat spesifiek vir 'n toepassing is, van die kernkomponente, wat generies is vir enige toepassing.

Die BASE argitektuur het gelei tot die ontwikkeling van die 3SAL aktiwiteitstruktuur om die kommunikasie en bestuur van industriële aktiwiteite in 'n digitale omgewing te vergemaklik.

Met die hulp van 'n bedryfsvennoot is twee gevallestudies ontwikkel om 'n implementering van die BASE argitektuur te evalueer. Die maatskappy is 'n lugvaart saamgestelde-mater komponent vervaardiger en is gekies vir die arbeidsintensiewe vereistes van die saamgestelde-mater bedryf. Die gevallestudies was daarop gemik om die argitektuur teen die drie geïdentifiseerde verantwoordelikhede van die administrasiedop te evalueer en te bepaal of die menslike werkers verhoog word tot holon status. Die eerste gevallestudie se doel was om te wys hoe BASE n koppelvlak tussen mense en digitale stelsels toelaat. Die eerste gevallestudie was ook 'n demonstrasie dat 'n toepassing in die bedryfsvenoot se fabriek toegepas kon word vir die tweede gevallestudie.

Die tweede gevallestudie het die effek geëvalueer wat BASE het op die Administratiewe Logistiek betrokke by die besigheidsprosesse waarby werkers betrokke was. Saam evalueer hierdie gevallestudies BASE se vermoë om die integrasie van mense in 'n I4.0 vervaardigingsomgewing te fasiliteer deur geïdentifiseerde verantwoordelikhede van die administrasiedop. Die evaluering het bevind dat die BASE HRH-AS die doeltreffendheid van Administratiewe Logistiek van besigheidsprosesse wat menslike werkers benut, asook nuwe geleenthede vir besluitneming op die winkelvloer, verbeter tot n mate wat voorheen nie moontlik was nie. Waarde-toevoeging, deur middel van die inpropkomponente van 'n BASE administrasie dop, is ook bewys deur die vermoë om outomatiese skedule-bestuur, outomatiese berekening van standaardwerk en verbeterde naspeurbaarheid met behulp van die 3SAL-aktiwiteitsstruktuur te doen.

*To my family and friends – "Anything is interesting if you look at it deeply enough. Study hard what interests you the most in the most undisciplined, irreverent and original manner possible."*

*— Richard Feynman*

# Acknowledgements

Firstly, I would like to thank Dr Karel Kruger. The way in which you guided my ideas, your depth of knowledge in our research field and your ability to distil the conclusions of our conversations means I owe the brevity of the architecture, and conciseness and thoroughness of this research to you.

To Professor Anton Basson, thank you for granting me the opportunity and funding to pursue this degree. Your experience and wisdom ensured that we always took a step back to evaluate the value and direction of our research, adding knowledge that extended past technical details.

Thanks, Reynaldo, for your help with the lab and out of the box questions during meetings that showed when there was an oversight or if further thought was needed.

Greg and Anro, you guys were always prepared to discuss ideas and share your experiences and knowledge. Being a year ahead of me you guys laid the foundation on how to approach this project. To the rest of the research group that joined us, your curiosity and energy to apply yourselves to your own visions spurred me on to make sure I lay down a platform for great things I could see us accomplishing together.

To my mom: You grew my interests and my mind, supporting my hobbies and guiding my early decisions knowing where I'd be able to flourish one day. I learned to grow excited about pursuits in knowledge, personal projects, and entrepreneurship as I saw you work to build your businesses and chase your interests. You taught me the value of taking action instead of waiting on chance.

To my dad: You're the reason I grew to love engineering. From the model plans you rolled out on the kitchen table you showed how ideas become reality through meticulous cutting and gluing transforming balsa into wings. Recently when I grew old enough to understand systems and processes, I gained an admiration for the work you do. It was a great inspiration to apply to this project what I learned about production from you.

Lastly, Monique, I would like to thank you for your love and effort that supported me during the long hours and frustrating debugging sessions. I would not have been able to pull this off without you (and thanks for the coffee machine).

# Table of Contents

# List of Figures

# List of Tables

**Page**

# List of Abbreviations

| | |
|---|---|
| **AR** | Augmented Reality |
| **ARTI** | Activity Resource Type Instance |
| **AL** | Administrative Logistics |
| **ALP** | Administrative Logistics Process |
| **ALF** | Administrative Logistics Function |
| **BC** | Business Card |
| **CPS** | Cyber-Physical System |
| **CSV** | Communicative State Variable |
| **CNP** | Contract Net Protocol |
| **CRUD** | Create Read Update Delete |
| **DMP** | Digital Messaging Platform |
| **DT** | Digital Twin |
| **ESV** | Environmental State Variable |
| **GUI** | Graphical User Interface |
| **HCPS** | Human Cyber-Physical System |
| **HMI** | Human Machine Interface |
| **HRH** | Human Resource Holon |
| **HRH-AS** | Human Resource Holon Administration Shell |
| **ICT** | Information and Communication Technologies |
| **IoT** | Internet of Things |
| **LUA** | Layup Activity |
| **OPC UA** | Open Platforms Communications Unified Architecture |
| **PSV** | Physical State Variable |
| **RAMI4.0** | Reference Architectural Model Industry 4.0 |
| **RFID** | Radio Frequency IDentification |
| **RFP** | Request For Proposal |
| **SBB** | State Blackboard |

| | |
|---|---|
| **SDL** | Service Description Language |
| **SPC** | Service Provision Contract |
| **TCP** | Transmission Control Protocol |
| **VR** | Virtual Reality |
| **WOI** | World Of Interest |

# 1 Introduction

This introduction gives the background and context to the dissertation, followed by the objectives and contributions this dissertation aimed to achieve. The motivation for conducting this research, as well as the methodology, are then presented.

## 1.1 Background

The manufacturing industry has seen three industrial revolutions and the fourth industrial revolution, also known as Industry 4.0 (I4.0), is currently in progress throughout the world. I4.0 represents a fundamental change to the way systems are managed using digitization and the internet, combined with machine learning and statistical analysis. I4.0 is driven by modern manufacturing requirements of flexibility, robustness to environmental changes, reconfigurability, and self-optimisation to match new custom requirements (Lasi, Fettke, Kemper, *et al.*, 2014; Geissbauer, Vedso & Schrauf, 2016).

While I4.0 research has predominantly focused on connecting machines and digital systems, it is important to consider the integration of humans within the I4.0 vision. Human workers currently offer irreplaceable, critical skillsets in the manufacturing environment, contributing to "roll-up-your-sleeves effort, resourcefulness and creativity" to achieve new levels of performance (Rother, 2010). Apart from their autonomy and problem-solving capabilities, humans possess dexterity, sensory abilities and decision-making ability that is yet to be matched by autonomous systems.

This dissertation presents the need for and value of a digital administration shell for human workers to facilitate their integration into Cyber-Physical Production Systems (CPPSs) of Industry 4.0. The proposed solution takes inspiration from the Digital Twin concept, Holonic Manufacturing Systems, and modern interfacing technologies.

The Mechatronics Automation and Design Research Group started with research on Reconfigurable Manufacturing Systems and proceeded to adopt the Holonic Manufacturing Systems philosophy (Kruger & Basson, 2015). Thereafter, group's research focus shifted to the development of Digital Twins for existing manufacturing resources (Redelinghuys, Basson & Kruger, 2018). The research group then identified the need to integrate human resources with the digital factory environment to the same degree as the machines of previous research projects, and the concept of a Human Digital Twin is explored as part of the research in this dissertation (Sparrow, Kruger & Basson, 2019).

This dissertation firstly presents a description of identified difficulties to human integration with I4.0 environments. It then discusses the requirement and functional analyses for a generic human administration shell aimed to address these difficulties. These requirements are then used to formulate an architecture to guide the development and implementation of such an administration shell. Metrics for evaluating the effectiveness of such an administration shell are created and used to test an implementation of the architecture in an industrial case study.

## 1.2 Objectives and contributions

The objective of the dissertation is to develop and evaluate an architecture for implementing a digital administration shell that will facilitate the integration of human workers into an Industry 4.0 environment. The administration shell aims to elevate human factory workers to Cyber-Physical System (CPS) level and qualify them to Human Resource Holon.

The dissertation also considers aspects of interfacing and state digitisation required to give the human presence in the cyber world. This involves how the administration shell will use modern interfacing technologies to intelligently deliver information, so that decision-making and safety can be improved in factories of the future.

Furthermore, a structured method to communicate activities between CPSs for cooperation and dynamic scheduling in an I4.0 environment is developed as part of the objectives of this research.

The contributions of this dissertation includes identifying and describing aspects of modern organisations that may hinder effective integration of humans with their digital environments. These aspects have been termed Administrative Logistics. Along with the identification and description of the effect Administrative Logistics have on effective human integration, an architecture for a digital administration shell that facilitates and automates these Administrative Logistics is developed. The administration shell elevates the human worker to CPS status and forms a Human Resource Holon.

The dissertation also formulates a set of metrics for measuring the effectiveness of Administrative Logistics in industrial settings. These metrics are applied in a case study to determine if an implementation of the administration shell improved the Administrative Logistics of business processes in a manufacturing environment.

## 1.3 Motivation

The processes of the world's top producers reflect the importance of humans in their production models. BMW has declared that full automation is not their goal, since a human is unbeatably flexible (Loveday, 2018), and reflection on Tesla's Model 3 production issues revealed that "excessive automation at Tesla was a mistake" (Musk, 2018). This view is also shared by the composites industry, where the nature of the materials and complexity of manufacturing tasks make automation challenging or even impossible. Here, the skills of human workers are critical to production (Sparrow, 2017).

Interviews with industry pointed out that manufacturing in developing countries relies on smaller volume production of a larger variety of products. I4.0 can play an important role in this context by reducing the reliance on expensive manufacturing equipment and enhancing the use of information technology for optimizing the shop floor. In effect, it could level the playing field between factories with expensive machines and those that can utilize their human workers' full potential (Murthy, 2018).

The motivation for this research came from the possibility of developing technology on the wave front of current technological developments with great potential to aid both the international and South African manufacturing industries. The development of human centred technology in Industry 4.0 has barely been explored, and this offered an opportunity to develop new technology that could potentially bridge the gap between industries with the funds to make use of expensive and smart automation, and industries that can make full use of the strengths of their labour force.

There also exists new opportunities related to the integration of interfacing technologies. Using augmented reality and modern interfacing technology, AGCO reported 50% reduction in learning time, 30% reduction in inspection time, 25% reduction in production time - similar advantages of using smart glasses were reported at DHL, which is one of the leading logistics companies in the world. These two results were discussed by Ruppert, Jaskó, Holczinger, *et al.* (2018). These human interfacing technologies show great potential in individual use cases but require a system that can orchestrate them to be used in the dynamic and complex factories of the future while adapting to their users.

It is believed that better integration of these kinds of technologies, with upcoming developments in intelligent factory management systems and the human workers in them, will provide major advantages in competitive manufacturing, safety and job security.

## 1.4 Methodology

Before an attempt could be made at what a worker administration shell would encompass, insight into the needs and requirements for Industry 4.0, as well as humans in a modern factory environment, had to be gained. A literature review was thus conducted on I4.0, its expectations and the developments that have been made to reach those expectations, along with the identified enabling technologies.

Research on humans in Holonic Manufacturing Systems is sparse and most authors only note the need for developments in this area. Further literature on the integration and interfacing of humans with a cyber world, from fields other than manufacturing, was reviewed in order to understand this problem and its possible existing solutions. A pin board of needs, expectations, available technologies, and expected future technologies was created from the literature reviewed, which was refined as greater insight was gained regarding the specific relationships between I4.0 components. Refinement involved conglomeration or creating more encapsulating terms where appropriate.

For an initial structure to the architecture, research was done on existing technologies that try to achieve a similar goal - with the best examples found in the gaming industry. A simplified model of a massively multiplayer online game character was used as the basis for structuring the human administration shell.

Certain key questions needed to be answered during the development of the architecture, as some foundational work for its existence had not been done. This involved a generalised I4.0 activity description (discussed in Chapter 5) and principles of interfacing with humans in an I4.0 environment.

When the architecture came to be implemented, a suitable case study (discussed in Chapter 9) was chosen in collaboration with industry to validate the architecture's real-world application and not just its compatibility with recent I4.0 literature. The case study involved the implementation of the architecture to meet existing needs and goals set by the collaborating industry for their workers in conducting an activity. Both the industry's take on the value that was added, as well as the value the architecture has to I4.0 research, is evaluated in Chapter 10.

## 1.5 Dissertation structure

The dissertation starts with a review of relevant literature in Chapter 2 which cover modern manufacturing paradigms such as cyber-physical systems, digital twins and holonic manufacturing systems. The literature review also covers research on humans in Industry 4.0 and what aspects of their integration into modern manufacturing requires attention.

Following the literature review, Chapter 3 discusses the first contribution of the dissertation – the concept of Administrative Logistics and how many of the problems with effective human integration stems from ineffective Administrative Logistics. A solution in the form of an administration shell is proposed and three responsibilities it should fulfil are identified based on the concept of Administrative Logistics in Chapter 4.

Chapter 5 discusses the second contribution as part of the developed architecture, the 3SAL activity structure. The 3SAL activity structure aims to federate and standardise the communication and handling of activities entities n a manufacturing environment are involved with.

Chapter 6 presents the primary contribution in the form of an architecture for implementing an administration shell that can perform the three identified responsibilities. An implementation of the core components of the architecture is presented in Chapter 7. Along with the primary architecture, the dissertation further discusses requirements for successful human interfacing by means of Interface Resource Holons in Chapter 8.

An implementation of the architecture, along with two case studies to evaluate its effectiveness, is then presented in Chapter 9. The evaluation of the architecture implementation, in the context of the two case studies, is presented in Chapter 10. Chapter 11 concludes the dissertation, offering a summary of the presented research and recommendations for future research.

# 2 Literature review

This chapter provides an overview of technologies and concepts involved with the requirements and development of the architecture presented in Chapter 6. First, an overview of Industry 4.0 is given. Section 2.1 discusses the driving forces of I4.0, while Section 2.2 discusses some of the most relevant enabling technologies for I4.0. Section 2.3 discusses humans in I4.0, while Section 2.4 discusses Holonic Manufacturing Systems and how humans fit into its existing research and architectures. To explore the nature of human integration with organisations and manufacturing, literature discussing the nature and administration of organisations was reviewed in Section 2.4.

## 2.1 Drivers of Industry 4.0

The fourth industrial revolution (I4.0) is driven by new requirements for manufacturing, namely: increased complexity of products and processes; more frequent product changes, reduced investment; and an increased emphasis on process robustness and adaptability (Schroeder, Steinmetz, Pereira, *et al.*, 2016; Lasi *et al.*, 2014). Modern developments in Information and Communication Technology (ICT) have been described as actively pushing the development of I4.0 by introducing new technology to processes designed around the limitations of previous technologies (Lasi *et al.*, 2014).

## 2.2 Enabling technologies for Industry 4.0

Industry 4.0 is essentially a revolution in how ICT is used to integrate and organise physical systems to be flexible and robust, while tending to modern manufacturing requirements. Since I4.0 is still in its infancy, research has mainly focussed on reference architectures to integrate and organise these new requirements and technologies. A notable example of I4.0 architectures is being developed by the German Electrical and Electronic Manufacturer's Association and their 600 associate companies, which they have called Reference Architecture Model for Industry 4.0 (RAMI4.0) (Schweichhart, 2016).

### 2.2.1 **Industry 4.0 and cyber-physical systems**

The capabilities of the physical world are expanded through embedding modern computational and sensory technologies. With the development of enabling technology (e.g. the Internet of Things, cloud-based services, inexpensive micro-computers, smart-phones, smart sensors, Wi-Fi, etc.), these digitally controlled physical systems can be described as CPSs and benefit from the ability to draw information from the environment around them and communicate with other CPSs as well as the internet (Baheti & Gill, 2011a). Due to the potential complexity and different ways CPSs can be implemented, reference architecture to guide their development have been created to try and ensure interoperability and structured development (Lee, Bagheri & Kao, 2015).

CPSs, with their connectedness, self-sensing, self-diagnosis and intelligent behaviour, can fulfil the requirements of I4.0. These systems will need to communicate amongst each other over an I4.0 compliant communication protocol.

The RAMI4.0 architecture describes a technology consisting of sensors and software that adds intelligence and connectivity to physical assets, enabling it to interact with other cyber systems - at which point it is elevated to a CPS and referred to as an “I4.0 component” (VDI, 2015). Applying CPS to production, Cyber-Physical Production Systems (CPPS) can enable manufacturing entities to reach resilience, self-adaptability and intelligence, and transform factories with significant economic potential by integrating logistics and services (Lee *et al.*, 2015). The RAMI4.0 administration shell concept is illustrated in Figure 1.

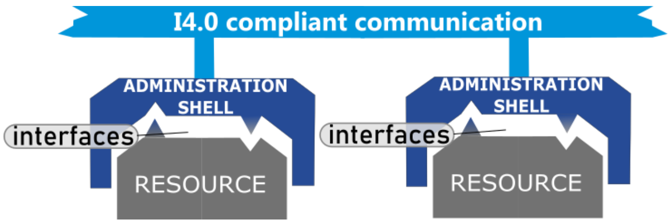



**Figure 1 – The RAMI 4.0 Administration Shell (reproduced from VDI (2015))**

Digital technologies provide the communication and information processing abilities required for I4.0, along with the increasing number of sensors and data acquisition, storage and processing mechanisms. While no concrete standard has been reached, existing communication protocols such as OPC UA, MQTT and Automation Mark-up Language (Automation-ML) are widely considered as strong bases for an I4.0 communication (Kritzinger, Karner, Traar, *et al.*, 2018; Redelinghuys *et al.*, 2018).

### 2.2.2 **Digital twin technology**

With the development of the *Internet of Things*, *Big Data* and machine learning, large volumes of data can be collected and aggregated throughout an entity’s life-cycle. This requires a structured method for handling the historical state, current state, and predicted future states of that entity. The *Digital Twin* (DT), a concept that was first mentioned by Grieves and Vickers (2016) simply as a "Conceptual Ideal for Product Life-cycle Management", is a technology developed to address this requirement. Figure 2 illustrates the DT concept.

Aaron and Lane (2017) defines the DT as "an evolving digital profile of the historical and current behaviour of a physical object or process that helps optimize business performance". The DT would allow the benefits of digital technologies, such as analysis on historical data, monitoring of the current status, and simulation for predicting future states of the system enabling advantages such as individualised maintenance, continuous improvements, and enhanced control.

Research on the realization of the DT has followed different approaches. Kritzinger *et al.* (2018) make an important distinction between the concepts of a *Digital Model*, a *Digital Shadow*, and a *Digital Twin*. A Digital Model is a static representation of an asset or product in digital format (e.g. CAD files, material specifications, etc.), while a Digital Shadow is an automatically updated digital representation of an asset or product instance to reflect the real-world system state. The Digital Twin extends the Digital Shadow by feeding back information to the real-world product or asset - augmenting its behaviour with the intelligence and knowledge from the cyber world. Figure 2 illustrates the feedback defining a Digital Twin from a Digital Shadow.

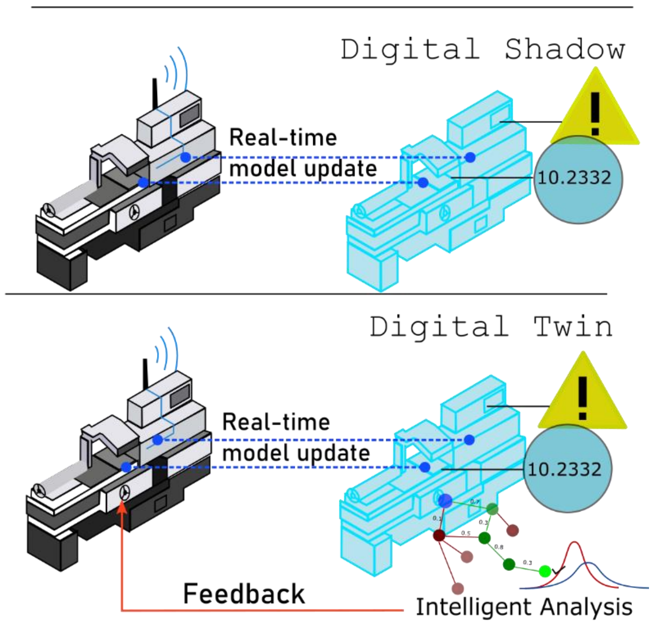



**Figure 2 – Illustrated Digital Shadow and Digital Twin (adapted from Kritzinger *et al*. (2018))**

### 2.2.3 Holonic Manufacturing Systems

Holonic Manufacturing Systems (HMS) are inspired by the concept of a holon, which is an autonomous, cooperating and goal-driven entity that itself forms part of a larger holon (Van Brussel, Wyns, Valckenaers, *et al.*, 1998). HMSs offer robustness, flexibility, scalability and reconfigurability due to their fractal nature, intelligent autonomy and cooperation with each other (Valckenaers & Van Brussel, 2016). A HMS applies this holonic concept to integrate all the manufacturing activities and resources of an enterprise.

The Reconfigurable Manufacturing System (RMS) paradigm is often closely linked with the HMS paradigm. RMSs are manufacturing systems that enable elements with similar functionality to be interchangeable with minimal delay or effort. RMSs place emphasis on flexibility, reconfigurability, and robustness (Kruger & Basson, 2015). With modern manufacturing systems, I4.0 manufacturing has seen prolific research interest in several manufacturing system philosophies in an attempt to address RMSs. Holonic, fractal, and bionic manufacturing systems are some of the more developed philosophies. Arguably the philosophy that encapsulates most of the functionality of fractal, organic, and bionic systems are holonic systems (Tharumarajah, Wells & Nemes, 1996).

A holon is distinguished by the following properties (Cardin, Derigent & Trentesaux, 2018; Valckenaers & Van Brussel, 2016):

- A holon is autonomous, cooperative, reusable, and self-configurable.
- A holon has a recursive structure (i.e. can consist of other holons, and form part of a larger holon).
- A holon delivers a specialised service along with other services from its peer holons to collaborate towards a common goal.

The internal structure of a holon in Figure 3, as proposed by Nylund, Salminen & Andersson (2008), takes into account technological developments in virtual modelling, simulation, and digital storage. This holon structure reflects the idea of a communication shell as in the RAMI4.0 architecture. A key part of any HMS is communication between holons, since without it the cooperative aspect of the system fails.

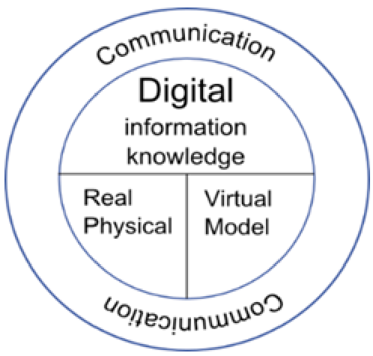



**Figure 3 – The structure of a holon (reproduced from Nylund et al. (2008))**

Research on HMSs have produced several reference architectures, with each identifying specific components and holon classes through separation of concerns. The most notable architectures are: PROSA (Van Brussel *et al.*, 1998; Leitão & Restivo, 2006), which tries to address some of the global optimisation issues not addressed by PROSA; and very recently, ARTI (Valckenaers & Van Brussel, 2016), which represents a generalisation and refinement of PROSA for broader application domains.

#### 2.2.3.1 PROSA

The Product-Resource-Order-Staff Architecture (PROSA) was introduced in 1995 by Van Brussel, Wyns, Valckenaers, *et al.* (1998). PROSA defined the interaction of identified specialised holon types that represent different components of a manufacturing process. The three basic holon types discussed by PROSA are:

- The Resource Holon, containing a physical part of the production system such a tool or material, and an information processing part controlling the resource.

- The Product Holon, holding process and quality knowledge that will allow the orchestration of Resource Holons to produce an instance of the product it represents.

- The Order Holon, a representation of a task in a manufacturing system where it orchestrates Resource Holons to fulfil the actions as stated by the Product Holon.

PROSA also specifies that data always belong to the holon that maintains the data. This compliments the distributed control methodology and improves scalability and reconfigurability since data the holon needs for its specialised decisions are always maintained and available to it.

Van Brussel, Wyns, Valckenaers, et al. (1998) also commented on the robustness of holonic systems as their fractal nature introduces stable intermediate forms on which the holarchy is expanded. In other words all components of the system do not have to be operational for the system to run stably at reduced functionality.

#### 2.2.3.2 ARTI

The Activity Resource Type Instance (ARTI) reference architecture was created by Valckenaers (2018) as a generalisation of PROSA to increase adoption of the architecture and increase flexibility. Valckenaers identified that early introductions of limitations and constraints on architectures cause problems in flexibility since they introduce legacy requirements on potentially unforeseen circumstances. Valckenaers gives the example of how distinguishing Resource

Holons from Transport holons reveals complications in software and development when the transport plays a resource role as part of a logistics operation.

ARTI makes an important distinction between generic, and specialised components in the architecture, separating all decision-making functionality into plugin components. ARTI identifies that legacy constraint issues arise from making specialised decision components part of the general solution, requiring many components of the system to be changed when the decision-making changes.

Valckenears goes on to discuss how a holarchy with such concurrent and modular architecture would see a real world implementation, reflecting on his experiences in using traditional object oriented programming languages before expressing the success their research obtained by using the highly concurrent, functional programming language, Erlang. Valckenears described the benefits arising from the separation of individual Erlang process states, the ability to represent thousands of light weight holonic components as processes inexpensively, and the stability and fault tolerance of the Erlang virtual machine.

## 2.3 The Resource Holon

The Resource Holon is a holon that provides any service towards the realisation of an enterprise goal. A service can be provision of physical materials, computing power, or knowledge. This section discusses the responsibilities a Resource Holon needs to fulfil to successfully take part in an HMS, as well as a description of the *World of Interest*, which refers to the parts and information of the world relevant to the Resource Holon. When a Resource Holon is added to the holarchy, it advertises its capabilities to the holarchy which prompts other holons to evaluate possible process improvements or new capabilities to fulfil orders (Van Brussel *et al.*, 1998)

### 2.3.1 **Resource Holon responsibilities**

To successfully form part of a manufacturing holarchy, the Resource Holon needs to comply to the responsibilities described by Valckenaers & Van Brussel (2016):

- **Reflection of reality**: The Resource Holon must reflect its corresponding resource, with synchronized information about state and expected future states, using knowledge about its dynamic behaviour.
- **Information provision:** Provide resource related information to other holons on its processes, local topology, and possible constraints.
- **Maintaining a local schedule:** Own an agenda that records future tasks to provide a reservation system and keep track of the availability of the resource over time.

- **Managing its local schedule:** The Resource Holon has local authority on how it accomplishes the scheduled tasks, which depends on the holon's capabilities.
- **Virtual execution:** This is a service for the holons responsible for logistics (e.g. order holons) that can ask for the virtual outcome of an operation (e.g. time or quality). Using its local schedule and reflection of reality, the Resource Holon should be able to provide this information.
- **Controlling the physical component:** The Resource Holon controls the real-world resource by starting and stopping its scheduled tasks and monitoring the execution.

### 2.3.2 The Resource Holon's World of Interest

The World of Interest (WOI), as described by Valckenaers & Van Brussel (2016), refers to the parts of the physical world that are relevant to an entity acting in it. Any information it can access and utilise in its decision-making will form part of the WOI. The holon acting in this WOI will need to reflect the WOI as a real-time updated digital model with the information relevant to it. The following points on the WOI are summarized from Valckenaers & Van Brussel (2016):

- The digital WOI should be a single source of truth to other entities dealing with the same WOI variables; therefore, information consolidation between holons with an overlapping WOI is essential.
- The WOI should be updated as the physical world changes to keep the reflection accurate and relevant for entities in the WOI.
- The rate of information update of the WOI will depend on the variables and how they are used - where a map of a building may only require an update rate of once a year, the position of an entity in that building may require an update every second.

The world-of-interest only needs to reflect reality to the point of allowing successful decisions to take place and Valckenears identifies the role that the DT as the source of data for this world-of-interest.

## 2.4 Humans in industry 4.0

The following section discusses literature concerning various aspects of humans in modern factories and I4.0 environments.

### 2.4.1 Human resources as decision resources

Literature identifies the critical decision making role that humans play in modern manufacturing environments. Pacaux-Lemoine *et al.* (2017) first identifies studies that show a rise in manufacturing and machine complexity has seen a higher rate of human errors occurring. Their research then explains that despite human being of superior intelligence to machines, the techno-centred approach to the design of the manufacturing systems involved in these errors come from the differences in how machines and humans receive, processes, and communicate data.

Pacaux-Lemoine *et al.* (2017) then discuss how a human centred approach to the design of intelligent manufacturing systems would address this issue. This research pointed out that modern manufacturing systems must have human awareness, while keeping human decision-making in the loop at different levels of automation. Human machine cooperation principles as well as different aspects of human responsibility and capabilities are explored in the suggested framework.

The concept of a Human DT arose from the success that DT technology experienced in manufacturing, and research was conducted to extend this concept to humans. Barricelli, Casiraghi, Gliozzo, et al. (2020) recently published research on managing human fitness through keeping a DT of the human updated by various biometric sensors. Their research focuses on the “human-in-the-loop” aspect that the Human DT technology can bring to various industries.

Literature identifies the importance of allowing humans to play to their strengths of handling unforeseen situations and creative problem solving, and modern manufacturing systems need to support these abilities (Woods, Tittle, Feil, *et al.*, 2004). The inability to act, control, or retrieve the right information can leave the human as powerless witness despite having the knowledge to solve a critical situation (Pacaux-Lemoine, et al., 2017). Pacaux-Lemoine *et al.* (2017) summarises from similar texts several rules for human centred design as:

- The human must always be aware of the situation.
- Avoid repetitive actions/decisions.
- Mental workload must be carefully regulated.

More recently, Ji, Yanhong, Baicun, et al. (2019) discuss the Human Cyber-Physical System (HCPS), taking into account that humans play an import part of the decision-making processes in CPSs and should not be seen as an external entity to the systems. They discuss a HCPS framework in terms of knowledge flow from the human through a cyber system to physical systems. The middle-man cyber system aids in analysis, sensing, and controlling to augment the human’s mental workload and effective decision making. The cyber system also provides the feedback from the physical systems, being able to intelligently control this feedback to best suit the operator.

### 2.4.2 **Human interfaces and human digital representation**

In order to allow humans the decision making and action taking abilities identified by literature in section 2.4.1, human-machine interfaces are required. Modern interfacing technologies have aimed to expand the flexibility of human inputs and maximise the use of human senses to make full use of their decision making capabilities (Bertram, Birtel, Quint, *et al.*, 2018) as well as explore new methods of training and process design (Loch, Ziegler & Vogel-heuser, 2018).

Human interfaces that can digitise commands or the human's state is not only important for human to machine communication but also improves human to human communication and orchestration. Digital messaging platforms and team management systems such as Asana or Microsoft Teams shows the benefits of providing a digital presence to human workers, but are still limited to inter-human interactions (Lopes, Oliveira & Costa, 2015).

Currently these human resource management platforms are still limited to desktop or tablet devices. For workers that do not need to be mobile or use their hands this may be adequate, however research on technology to address these limitations has been conducted. Ubiquitous Computing or Ambient Intelligence Environments aim to produce technology with interfacing capabilities that adapt to the situation and individual's requirements (Riva & Vatalaro, 2014; Weiser, Gold & Brown, 1999).

### 2.4.3 **Human-machine cooperation**

Humans in modern manufacturing environments oversee machines that make simpler and more repetitive decisions. An important balance between control, authority, and ability needs to be kept between humans and modern intelligent machines (Flemisch, Heesen, Hesse, *et al.*, 2012). Decisions to adjust the levels of control and authority based on abilities of the human and machine require effective communication and orchestration between the two.

Exploring the issue of human integration, Rey, Carvalho & Trentesaux (2013) report that careful consideration needs to be given to the difference in how artificial systems and humans treat response times, background knowledge, context understanding, information management and other cognitive functions. The authors then propose different ways in which humans and Artificial Self Organising (ASO) systems can exploit mutual advantages, and discuss Human-ASO integration issues – specifically considering tight timeframe requirements for decision-making on large amounts of data and calculations, and where communication breaks down due to a human's nature to summarize data into abstract ideas.

The HCPS concept is accompanied by a similar concept, *Operator 4.0*, which stems from the same realisation that humans and technology form a symbiosis in the manufacturing environment (Ruppert *et al.*, 2018). Romero, Stahre, Wuest, *et al.* (2016) identified eight augmentations for I4.0 operators that will form a HCPS, as presented in Figure 4.

Operator 4.0

| | | | | |
|---|---|---|---|---|
| Augmentation | Exoskeleton | Wearables | Virtual Reality | Intelligent Personal Assistent |
| Operator/HCPS type | Super-Strength Operator | Healthy Operator | Virtual Operator | Smarter Operator |
| Augmentation | Colaborative robot | Social networks | Big data analysis | Augmented Reality |
| Operator/HCPS type | Collaborative Operator | Social Operator | Analytical Operator | Augmented operator |

**Figure 4 – Operator 4.0 and HCPS (adapted from Romero *et al.* (2016))**

The enabling technologies and design principles for Operator 4.0 solutions are discussed by Ruppert *et al.* (2018). Peruzzini, Grandi & Pellicciari (2019) describe an overall framework for the Operator 4.0 approach - identifying the integration of human interfaces needs *human centred desi*gn, as was discussed in Section 2.4.1, along with human factors engineering.

Burns, Manganelli, Wollman, et al. (2018) discuss how the human aspect of the National Institute of Standards and Technology (NIST) Cyber-Physical Systems framework needs to be elaborated on. In their paper an example is given to show how smart sensors in a collaborative and interconnected smart city environment facilitates human safety and quality of life aspects. These principles not only apply to smart cities but to the safety and collaboration of humans and machines in factories.

Human interfacing with a digital environment that accommodates a person’s mobility and personal preferences have seen the development of Ubiquitous Computing environments, also called Ambient Intelligence Environments. Riva & Vatalaro (2014) lays out components and relationships that may aid in seeing the Ambient Intelligence Environment vision to reality. Due to recent developments in micro computing technology and smart materials, the Internet of Things is suspected to be a prime enabler for these Ambient Intelligence Environments (Lee, 2017).

### 2.4.4 **Humans as entities in an organisation’s structure**

To understand the role humans have in an organisation among each other and their machine peers, it is important to understand the structures and working of an organisation from a first principle basis. Herbert Simon describes the relationship between organisational entities, structure, and roles of administrative behaviour in orchestrating and influencing entities to achieve an organisation's goals. Understanding these concepts will be key to understanding human integration (Simon, 1997).

Simon identifies four modes of influence that affect an entity's behaviour in an organisation. These modes of influence include authority, training, advice, and the pursuit of efficiency. Entities reacting to these modes of influence may exert influence on other entities or perform some activity that they are responsible for.

## 2.5 Humans in Holonic Manufacturing Systems

In terms of the ARTI holonic architecture, a holon involving a human would be a Resource Holon, as the human provides services realizing the goals of a Logistics Holon (Valckenaers & Van Brussel, 2016). Due to the benefits of developing I4.0 components with holonic principles, attention is given to humans in HMSs and how humans would integrate with HMSs.

PROSA requires that the resource holon maintains data on its capabilities (list of products), its running tasks, its sub resources, and a log of its activities (Van Brussel, Wyns, Valckenaers, et al., 1998). The product holon holds a process plan, a product description, and the quality requirements. The order holon keeps track of the state of the physical product, the progress of the task, and historical data of the tasks.

PROSA separates holon types through specialisation. This separation saw humans as a specialisation in itself (staff holons) and did not consider the roles humans could play as resources as peers to machines. In PROSA's refinement, ARTI, humans can fall under a further specialisation of Resource Holon along with the activities they can perform (Valckenaers, 2018).

Even though humans now had the possibility of playing a more integrated role in an ARTI holarchy, Valckenaers (2018) pointed out that humans are activity *performers* and not resource holons - lacking the presence in the digital context that fellow holonic entities would have. Their shortcoming as a resource in a holonic manufacturing system stems directly from their inability to communicate and handle data as other digital holons would. Valckenaers (2018) also mention the need for a digital extension to the human for them to perform at the level of other digital holons, similar to the concept of the RAMI4.0 administration shell.

Paulo, Ramos & Neves (2007) noted that the promise of holonic manufacturing to effectively integrate humans is still to be realised, as no real-world implementations have been done - most research is focused on the automation of processes and not integration of humans. This promise was based on the holonic concept being derived from observation of how human societies self-organise and co-operate, meaning humans will be able to relate to holonic entities more than other types of artificial entities. They also suggested that artificial holons will require bidirectional *Person-Machine Interfaces* to give information and commands to the operator, as well as gather input and commands from the operator.

## 2.6 Conclusion

Human workers are still critically required in production systems due to their unmatched flexibility, robustness, and intelligence, despite their limits in strength and precision compared to machines. New ways to integrate humans and their strengths with other CPSs is needed to ensure these strengths are fully realised in modern factories. Operator 4.0 and the HCPS are recent developments in I4.0 research to explore these issues.

I4.0 is a paradigm shift in how business systems are structured and managed with the use of modern information and communication. Industry 4.0 developed as the result of trying to address the newly placed emphasis on customization, optimisation, and robustness in a changing environment requiring production systems to be flexible and adaptive. It was found that there is a lack of research on the integration of human workers into an I4.0 environment, and the challenges they will face with interfacing with these new technologies.

Reconfigurable Manufacturing Systems aims to address similar needs to that of I4.0 – with HMSs reported as a popular framework for implementation. These systems are enabled by various developments in computational, sensory, and communications technologies. The fusion of computational and communication elements with the physical world gives rise to CPSs which brings intelligence and communication to manufacturing components. Interoperability and convertibility are essential to the vision of RMSs and holonic systems. Although literature on holonic systems mention the need for research on how humans will fit into a digitally based HMS, very little has been done in this regard.

McFarlane & Bussmann (2003) mention the integration of humans by a human interface block, and various other authors stress the importance for human integration with HMSs. However, the serious communication and data processing differences between humans and digital holons have not yet been adequately addressed.

# 3 Administrative Logistics and the effect on human integration

This section will describe a concept defined as *Administrative Logistics* and how it affects the integration and organisation of entities in an enterprise. Administration, as used here, is discussed in depth by Herbert Simon in his book *Administrative Behaviour: A Study of Decision-Making Processes in Administrative Organizations* (Simon, 1997).

This chapter starts by explaining, on a first principles basis, the administrative functions that need to be facilitated by the digital administration shell for a human resource holon, and how these functions are critical to the successful integration of humans into the modern manufacturing environment. This chapter serves as foundation for developing the requirements of the HRH-AS, as well as metrics for determining the effectiveness of an implementation in Chapter 10, which develops a set of metrics for measuring Administrative Logistics effectiveness for a business process.

## 3.1 Organisational Administration

Simon (1997) describes an organisation as performing two roles: making decisions and "getting things done". Organisational administration, as described by Simon, involves the decisions, structure, and influences on organisational entities.

### 3.1.1 **Organisational entities**

This dissertation makes a distinction between two kinds of organisational entities in a manufacturing environment, based on their cognitive abilities: *influential* entities, and *non-influential* entities. In manufacturing, the result of administration is to exert a physical effect on the world through the influence of action. Influential entities are then the entities that can perform influential actions. These influential entities will also be referred to as decision makers or agents (workers and decision-making software), due to their ability to process information and influence other entities. In contrast, non-influential entities cannot make decisions or influence other entities (e.g. tools, materials, rooms, etc.). Typically, in manufacturing, a sequence of decisions end with a final influence on a non-influential entity.

### 3.1.2 **Organisational structure**

Simon's description of organisational structure is illustrated in Figure 5. Simon explains that organisational structure arises from decisions about the relationships between organisational entities, based on the influence they have on each other. Simon identified two general dimensions present in all organisations: a division of labour (vertical hierarchy) and a division of work (horizontal hierarchy). Where division of work allows specialisation and optimisation of skills and knowledge, division of labour allows the coordination of these skills to achieve a goal.

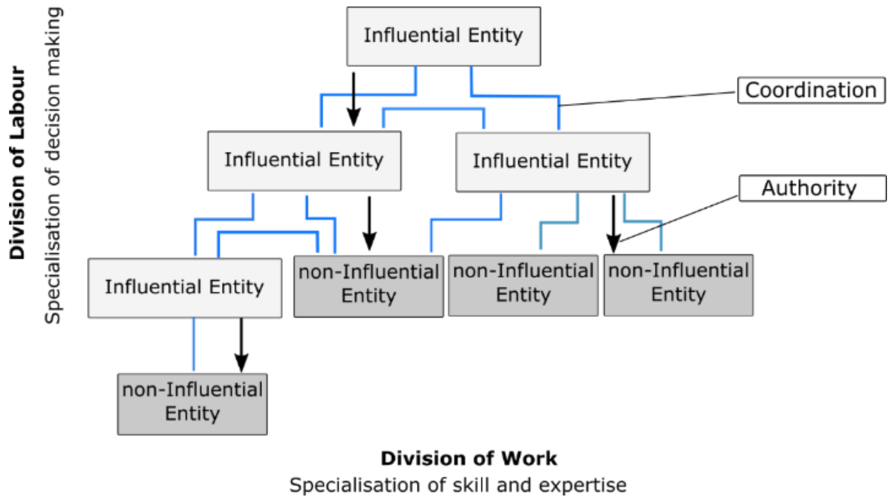



**Figure 5 – Illustrating organisational structure as described by Simon (1997)**

Figure 6 describes these concepts by Simon in terms of an OWL (Web Ontology Language) ontology for easier reference throughout this document. This figure aims to visually clarify the concepts, distinctions and relationships described in this section.

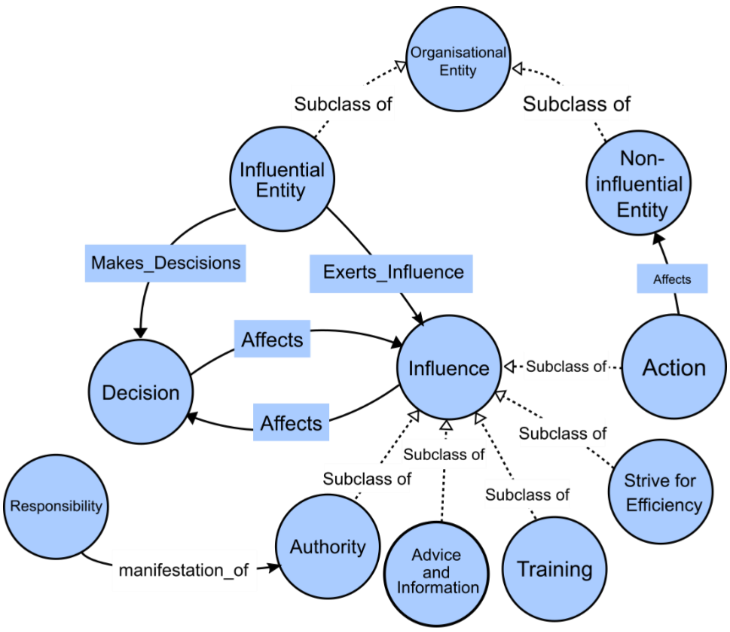



**Figure 6 – Ontology for organisational terms based on Herbert Simon's work in administrative behaviour.**

### 3.1.3 **Influence within Organizational Administration**

Simon identified four means of influencing an agent's decision making. These are *responsibility*, *efficiency*, *training* and *advice and information*. Considering the manufacturing context, one more refined subclass of influence is added – *action*. These means of influence (as represented in Figure 6) are described in the following sections.

#### 3.1.3.1 Responsibility

Agents are assigned responsibilities by other agents with authority. This responsibility is given by the authoritative agent as part of decisions it made to fulfil its own responsibilities. It should be noted that authority here does not refer to totalitarian control, but rather a willing reduction in the amount of choices an agent has for actions – allowing the agent to specialise their decision making.

#### 3.1.3.2 Efficiency

Simon recognised that there is an inherent drive for efficiency in people. This drive results in taking the shorted path, finding the easiest way to accomplish a task or minimising the length of an email. Machines are very far from being as good with this mode of influence as humans are, but are often programmed for optimisation and minimising cost. Since this means of influence is very particular to context, and is an internal influence, it will not be further discussed.

#### 3.1.3.3 Training

Agents are trained to internalise certain responsibilities and automatically act on them. Training automates how authoritative agents expect decisions to be made in certain circumstances and allows for specialisation of agents to the organisation's needs and the agent's capabilities. It is considered here that training encompasses the programming of machines, as well as human resources.

#### 3.1.3.4 Advice and Information

Agents busy with a particular task can be given advice from an agent with expert knowledge, as well as information on the environment or other agents to aid its decision-making processes. This advice or information can also be a mandatory signal to act on a given responsibility. For example, a person could be given the responsibility to act when receiving a certain piece of information.

#### 3.1.3.5 Action

Considering the manufacturing context, one more refined subclass of influence is added – *action*, or the manifestation of what Simon describes as "getting things done". This form of influence is exerted by an influential entity on a non-influential entity. Examples include working a piece of material, switching on a light, typing a document into a word processor, or handling a mop. This form of influence is typically the end product of administration in the manufacturing context.

## 3.2 Administrative Cognition

This dissertation identifies two distinct aspects of organisational administration, namely Administrative Cognition and Administrative Logistics. Administrative Cognition is any process that involves decision making, control, analysis, intelligence, or value addition to the organisation. Influential entities perform administrative cognition as part of fulfilling or creating responsibilities.

Part of the output of Administrative Cognition is influence on the organisational structure. This is the hierarchy of influence, communication channels and protocols. Managers determine the organisation's structure below them, and their own influences are structured by management above them. Closer to the shop floor, control on the structure becomes increasingly distributed and dynamic.

Administrative Cognition occurs through the flow of information or data into, and out of, decision making entities. How this information is moved, how it gets to the right entities, how it is collected and stored, and how agents find information they are looking for, is the process of Administrative Logistics, discussed in Section 3.3.

## 3.3 Administrative Logistics

For an organisational entity to take part in Administrative Cognition, it needs to be influenced by receiving responsibility, advice, or information from another entity. Administrative Logistics (AL) are any procedures and mechanisms used to deliver these modes of influence. Messaging, programming, uploading, talking, or writing on a notice board are all processes that fulfil the AL that one entity needs to exert its influence on another. Industry 4.0 is essentially a revolution in this aspect of organisational operations– driven by developments in information and communication technology and advancements in theories on distributed control, communication protocols and data science.

Common difficulties in organisations arise from allocating responsibility, changing relationships, and competing authority as visualised in Figure 7. Often the best method organisations have to deal with these difficulties is to create more vertical hierarchies, which are intended to resolve these conflicts only to be stifled by the bureaucracy associated with the increased AL.

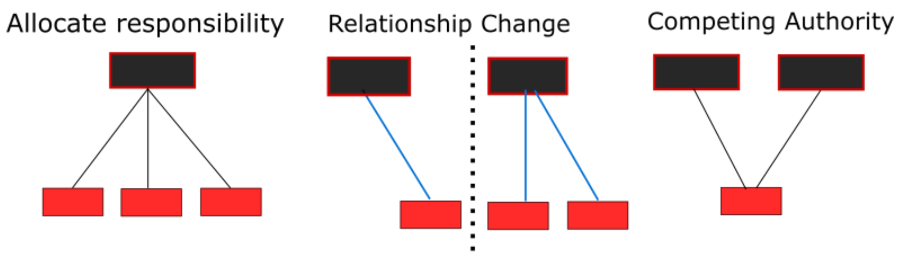



**Figure 7 – Organisational structure where inadequate AL can cause conflict.**

Different schools of organisational structure have traditionally been created to address a wide range of flexibility and robustness needs, without actively changing the organisation's structure. RACI[1] or other responsibility matrices describe some of these relationships (International Institute of Business Analysis, 2006). These techniques aim to mitigate and manage the cost of changing AL, such as changing offices, phone lines, post boxes, name tags, file processing procedures, etc.

Figure 8 shows a visual interpretation of organisational disarray resulting from wrong communication channels, programs searching for data in the wrong places, wrong activities performed out of habit, retraining times, etc. This disarray decays over time as the AL become more cemented into processes and people form new habits.

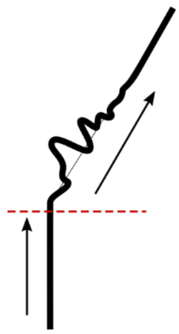


**Figure 8 – Organisational disarray after changes due to ineffective AL**

An organisational structure may work well in one context and fail in another, and with organisations pressured to be more flexible and agile in I4.0, effective AL becomes a critical factor to achieve this.

## 3.4 Human integration through supported and automated Administrative Logistics

Organisations are in a constant battle between having the most efficient structure for a particular goal, and the cost of re-organising its structure and the associated

[1] RACI is an acronym for derived from four key responsibilities typically found in an organization: Responsible, Accountable, Consulted, and Informed

processes that allow effective AL for the new structure. The set of actions and tools used to achieve a particular AL requirement is referred to as an Administrative Logistics Process (ALP). Remembering a value, walking to a notice board and updating a value written on it is an ALP, typing the new value into a digital notice board's user interface is also an ALP that may serve the same purpose, allowing the same influence to be exerted on the same entities. The one ALP could be said to be more effective than the other.

To visualise the benefit of effective ALPs, the reader only must imagine cutting out any form of digital communication, such as email, in a large organisation. This ALP will need to be replaced by other ALPs to transport information between members. Even if the same command can be written on paper and carried to another person in a building, this will have detrimental effects on real-time decision-making abilities and information quality and availability.

Recently, technologies such as Slack, Asana, Git, and other digital management software as well as lessons learnt from lean processing and lean start-ups, directly address the agility of AL. The successful management of teams is directly supported by these technologies due to improvement on communication channels and the availability provided by cloud hosted solutions.

I4.0 with holonic and service-oriented manufacturing systems are an attempt to create manufacturing systems with a less monolithic structure. Modularisation and self-organisation are an attempt to automate and simplify the AL involved.

Holonic design principles address this issue by allowing organisational entities to dynamically re-arrange themselves to match the requirements of the situation. Placing the responsibility of managing data and schedules on the individual holons. Holonic design principles aim to make the cost of effective AL insignificant, allowing distributed control and real-time decisions about problems at the source instead of waiting for commands to make their way up and down a vertical hierarchy.

Industry 4.0, with technologies such as IoT, digital twins, and the RAMI4.0 administration shell, can often be thought of to bring usually non-influential entities alive with the ability to from part of the business decisions around them. The data captured and managed by these technologies automate the previously impossible AL which would have been manually accomplished. With the ability of machines to take part in business decisions, their AL need to match those of their human peers.

Typically, the ALPs of machines and humans differ to the point of causing conflict. Incompatible and ineffective AL makes humans lose their ability to effectively influence other entities around them, despite having superior cognitive abilities. At this same time machines cannot influence humans to either to ask for assistance, warn, or give advice. The AL between humans and machines need to be addressed, and humans need to take full advantage of the benefits digitisation has to offer since this will benefit them in ALPs with other humans too.

Figure 9 maps the functional aspects of administration, and the properties that will determine its effectiveness. The relationships in this diagram is explained in the following sections.

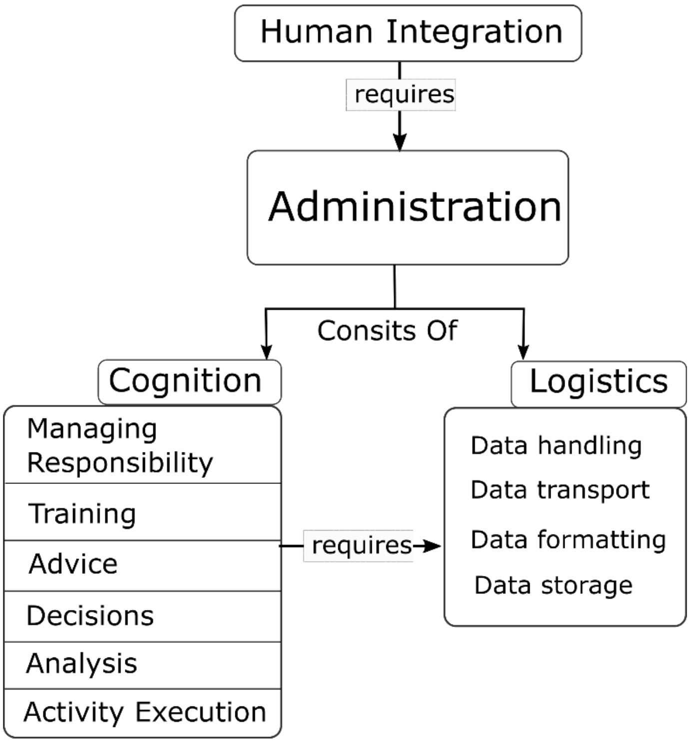



**Figure 9 – Administrative functions that human integration relies on**

From the concept of AL, the following sections discuss three responsibilities that an HRH-AS needs to fulfil to facilitate in ALPs involved in business processes.

## 3.5 Delegation as an Administrative Logistics Process

Entities in an organisation often need a service delivered from another entity to fulfil their own responsibilities. These services can be as simple as requesting advice or information, or involve asking them to perform some required activity. These decisions involve a dialog between an authoritative (responsibility giving) and subordinate (responsibility receiving) entities. The decisions and communication involved in this process forms part of the delegation responsibility of the HRH-AS.

### 3.5.1 Examples of common delegation situations

The following paragraphs present short examples of delegation scenarios and the AL involved.

- A manager has been given the responsibility to make sure some product is produced. The manager needs to delegate responsibilities to workers, and a dialog will take place based on the workers' schedules, their capabilities, and priorities. The exchange of information before the final hand-over of responsibility is the process of delegation.

- A more generic example where information is delegated: If an entity A requests information from entity B, B can agree to supply this data in which case it has accepted responsibility to supply the data. If component A requires component B to perform an activity, component B can accept this task and therefore has been given the responsibility to complete the activity. Decisions need to be made by the subordinate entity on whether it has the time, capabilities, willingness, or training to accept this responsibility.

### 3.5.2 Generalising delegation through the contract net protocol

Based on the examples in Section 3.5.1, delegation can be generalised through the Contract Net Protocol (CNP)[2] model. Although associated with agent-based systems, humans participate in CNPs mentally in all organisational exchanges without explicitly being aware. Figure 10 illustrates this generalised view of delegation – indicating the multiple non-value adding, yet critical, steps in this process.

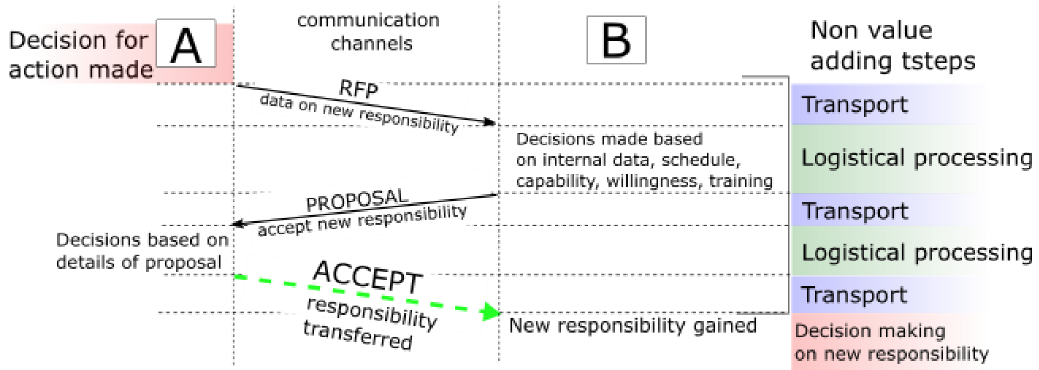



**Figure 10 – Delegation broken into its CNP components.**

Many instances of delegation can be automated for trivial responsibilities, such as obtaining information about a person's schedule, current activity, or stored

[2] The Contract Net Protocol is specified by the Foundation for Intelligent Physical Agents (FIPA) to deal with task sharing in multi-agent systems

information. It would benefit human workers to minimise their time spent on trivial delegations in an environment with many other entities and maximise their decision making and execution time on value adding tasks.

Another example of an entity acquiring a responsibility is the process of dealing with a detected event. This could range from noticing fires, broken components, or errors in processes to simpler events like noticing something needs to be cleaned. Figure 11 shows a typical process of event handling in an organisation where one entity specialises in the detection of an event and needs to give this event data to an appropriate peer entity for decision making and action taking.

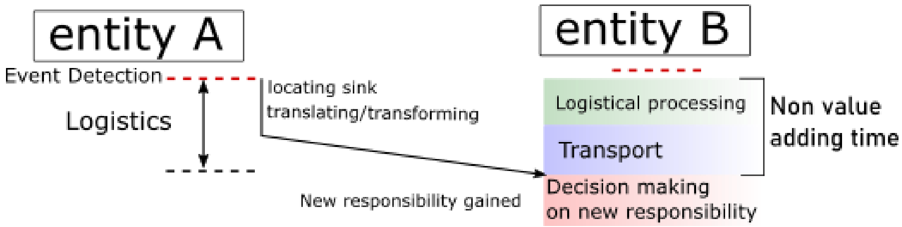



**Figure 11 – Event handling logistics**

In many organisations, critical events (e.g. machine failures or workplace hazards) have a fixed receiving entity and all other entities are trained (internalised) many of the AL components represented in Figure 11 (who to report accidents to, where to submit the report, etc). Any unforeseen events typically suffer from slow and ineffective AL. Figure 11 illustrates the AL components and time spent before decision making on event data can start.

## 3.6 Digital data processing and management:

Digital storage offers unmatched performance when it comes to portability, data lifespan, data volume and logistical processing such as translation, sorting and searching. It is necessary for the HRH-AS to manage digital data storage and processing requirements of a human on their behalf.

Keeping track of the human's schedule, sorting incoming messages from other holons, prioritising these messages, keeping a digital record of the humans capabilities, permissions, and statistics are all examples of data that needs to be processed, communicated, or stored.

In some instances, the data managed by the administration shell is there to augment and improve the workers internal training and activity execution. Instruction manuals, procedures, and activity specific information which can expand their capability and confidence. Herbert Simon stated that:

> *"The individual is limited by those skills, habits, and reflexes which are no longer in the realm of the conscious. The individual is limited by the extent of his knowledge of things relevant to his job. Making these skills or information available to the individual expands their capabilities, improving flexibility…"* (Simon, 1997).

## 3.7 Interfacing

As discussed by Sparrow et al. (2020), interfaces to humans in modern manufacturing environments will need to be adaptable and dynamic. Connecting to the most appropriate interface for the activity or situation at hand ensures robust communication between the human and the digital environment around him.

This also presents as an AL challenge that would follow a CNP model as shown in Figure 10 in order to select and connect with the most appropriate interface for the situation. The flow of information to and from the human is discussed in the following two sections, as the responsibility of interfacing that is allocated to the administration shell.

### 3.7.1 **Improve delivering information to the human**

Effective feedback or advice may be available to a worker but delivering this information quickly and effectively is the only way to realise this improvement. The feedback information exists, the worker that needs to consume this data exists, yet the AL often fail this operation in companies.

A new worker may not know his way around a factory, and the information he requires is available somewhere in the factory in the form of a map, but the AL of finding and delivering the information prevents or severely delays the worker.

### 3.7.2 **Improve obtaining information from the human**

A positive or important decision could be made by a worker, and the effects will not realise if the information was not delivered to the right people or machines, in a manner that they understood, in time. A worker may see a robot that will soon be in trouble, but an inability to communicate to the robot and be understood means AL has failed the realisation of this decision. A robot may need a person to move something out of its way or get out of its work area but have no way to communicate this information due to ineffective AL.

# 4 The Human Resource Holon Administration Shell

The Operator 4.0 and HCPSs concepts both require that human workers have a digital component to augment their shortcomings when interacting with a digital environment. The holonic framework offers a method for intelligent actors with different abilities to cooperate effectively. This dissertation presents a possible solution to these needs in the form of an administration shell, as described by the RAMI4.0 framework. This Human Resource Holon Administration shell (HRH-AS) will raise humans to a CPS level and facilitate worker interaction with other CPSs to effectively elevate the worker to Resource Holon status according to the ARTI reference architecture (Valckenaers & Van Brussel, 2016). I4.0 enabling technologies, the use of holonic manufacturing principles, along with existing human interfacing technologies and principles reviewed here, would allow such an administration shell to become reality.

The following sections discuss three identified responsibilities and human resource modelling that the HRH-AS will need to fulfil. With these responsibilities the physical human, together with their administration shell, can seamlessly integrate into the cyber-physical factory environment and qualify as a Human Resource Holon.

## 4.1 The HRH administration shell responsibilities

The administration shell needs to supplement the human's ability to store, process, and communicate relevant information of itself, and of services it can deliver, to other I4.0 components by making use of modern Human-Machine Interfaces (HMIs), the Human DT, and ICT. The proposed HRH-AS is structured around the ARTI holonic reference architecture to ensure the benefits from this architecture, such as integrability and modularity, is gained. Figure 12 illustrates the three identified responsibilities, which are discussed in the next sections.

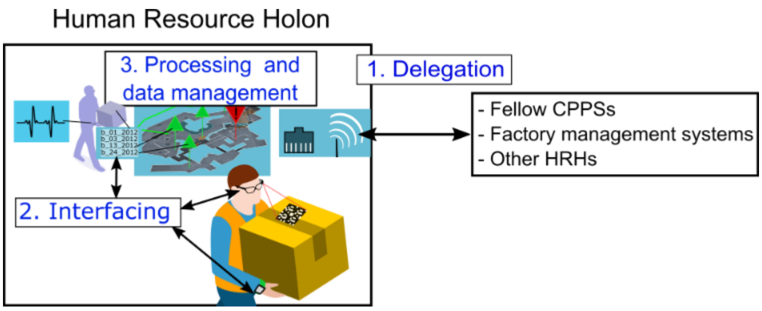



**Figure 12 – HRH-AS responsibilities**

### 4.1.1 Delegation on behalf of the human

Communication between holonic entities often involves simple data exchanges, such as the CNP or querying for state and schedule information. In these cases, effective AL is critical and humans in these holonic environments will need to take part in these communications to form an effective part of the holarchy.

Although humans are fully capable of answering questions from other holons through traditional Human-Machine Interfaces (HMIs), such as keyboards or touch screens, these conversations mainly involve exchanging numerical values or simple yes/no statements at digital speeds and precisions - neither of which humans are good at. The human thus becomes a bottleneck in the communication.

To avoid this bottleneck, the HRH-AS will need to play a delegation role on behalf of the human, using its knowledge of the human’s physical and mental behaviour and the human's schedule. While it should remain possible for a human to answer some more complex questions about his work or himself, there are benefits to automating parts of this process.

Humans are excellent at speculating on the outcomes of activities surrounding them and even the broader effect they will have on the world around them. They cannot calculate the outcome with decimal precision, but are able to draw from a much larger body of heuristics, experience and intuition than any computer. The human may be consulted by the administration shell if it cannot answer a question itself.

This administrative role will minimise interruptions to the human worker, who would otherwise have had to handle all incoming requests and recall all required data while trying to perform their own responsibilities. To use a hypothetical example demonstrating this delegation role (inspired by a similar scenario from (Van Brussel *et al.*, 1998):

An Autonomous Guided Vehicle (AGV) fleet, conveyor, and human can all carry a pallet across the shop floor and a logistics holon (e.g. an Order or Activity Instance holon) needs to choose one of the three resources to do the job. The AGV fleet and conveyor holons can instantly communicate the lead time, the energy cost, the path to be taken and any number of other details. In contrast, the human worker would need to manually calculate and enter this data on a keyboard, which not only interferes with his current work, but delays the decision-making process. The human worker is also burdened with calculating the details of accomplishing this simple task in order to give the logistics holon an estimate (Van Brussel *et al.*, 1998).

With the data held by the HRH-AS (such as the worker’s average walking speed) and a simple path-finding calculation, along with knowledge of the human’s current schedule, an estimated lead time can be calculated and sent to the logistics holon instantly. This leaves the human worker free to continue with their work, only to receive a notification from their administration shell to perform the task if selected. Of course, the human may refuse to perform the task, in which case the logistics holon will need to choose a different resource.

### 4.1.2 **Facilitate Human Interfacing**

The administration shell of the HRH will need to interface with its human to gather and deliver data. The data it gathers can be automatic, such as motion tracking, or manual, such as a voice command or touch interface. Some modern technologies that the HRH may utilise for receiving information from its physical human is RFID, GPS, Voice Command, Mo-Cap and eye-tracking.

Human interfaces provide two functions: observation and semiosis. Observation obtains any form of information or data from a human while semiosis delivers information to a human through stimulation of their senses. These concepts are discussed in more detail in Chapter 8.

For communication to the human, smart glasses, work station projection, interactive stores, haptic feedback and other emerging technologies will integrate the human with the cyber aspects of the manufacturing environment (Sparrow *et al.*, 2019). Bi-directional communication forms part of many emerging interface technologies, such as smart watches and touch screens, and will need to be utilised by the HRH-AS appropriately.

This orchestration of appropriate HMI technologies will be facilitated by the HRH-AS. Using the HRH-AS enables a high level of robustness and optimisation to be achieved – especially when used within an environment where the HRH-AS deals with holonic interfaces (Sparrow *et al.*, 2020).

### 4.1.3 **Digital Processing and Data management Augmentation**

Humans process information through a complex mixture of abstraction, pattern matching, heuristics, imagination and other means. Digital holons may compute and share data in floating point precision, communicate events with statistics, and work in times accurate down to milliseconds or less. Humans are limited to reaction times on the order of 200 ms, which is drastically increased when decision-making is involved (Hedge, 2013). These differences develop into fundamental challenges for the integration of humans with digital systems.

The average human lacks the ability to remember any significant amount of numerical data, sequences or strings of characters, unless the human spends time to store the values in long term memory. Short term memory is limited to 7 ± 2 items or concepts that can be held and three that can be worked with – and once concentration is lost, the chance of recollection is seriously diminished (Sanders & McCormick, 1993). Long term memory is also subject to deviation over time, and old memories are susceptible to suggestion (Zaragoza & Mitchell, 1996).

Identifying their own shortcomings in processing information, humans have long augmented their own abilities with tools such as the abacus, hourglass, diaries, post-it notes, and, recently, the culmination of many of these tools in the smartphone.

All the different methods in which humans have augmented their abilities in memory, processing, scheduling and reflection have generally been used as stand-alone items integrated only in the human's mind. The proposed architecture aims to provide a means to combine and integrate existing and future augmentations, simplifying and facilitating the AL involved with sharing this information to other decision-making entities.

## 4.2 Human modelling for supporting the three HRH-AS responsibilities

### 4.2.1 Function of the human model

To perform the three responsibilities mentioned in Section 4.1 effectively, an accurate and up to date digital model of the human worker should form part of the administration shell and its decision-making processes. Some functionalities the human model will be able to aid in is:

- To answer hypothetical questions on lead times, future HRH state, and behaviour for given conditions.
- To identify, avoid, and mitigate safety critical situations.
- To personalise and optimise interfacing devices and tools for the human according to their abilities, preferences, and current physical state.
- To keep track of future and past activities and events.

Human modelling has been researched extensively by different fields such as medicine, psychology, physiology, philosophy and neuroscience, and the integration of this research with modern I4.0 technologies will support the above functions and enable other benefits. While all these fields describe different parts of the same dynamic human system and can technically be unified into an all-encompassing theory of everything human, this is not a practical approach for a manufacturing environment.

A human's stamina graph on a construction site will be good enough and a molecular simulation of the ATP synthase is probably not required. Some behavioural modelling may be important, while real-time in-depth psychoanalysis may not be. An effective digital model of a human worker will vary in complexity and structure depending on the application, cost and processing power.

This dissertation suggests, as a starting point, three domains of modelling with an attempt at maximizing the orthogonality of concerns and minimising overlap. The three modelling domains are described in the following sections.

### 4.2.2 **Kinematic modelling**

Technology for kinematic modelling has been in development by various industries for several years. Arguably the most well-developed is from the gaming industry. A glimpse at work from Valve, Epic, and independent developers reveals mature skeletal animation, motion capture, collision handling, condition-based events, real-time physics, reverse kinematics, path finding and various other technologies. Automation-ML has adopted COLLADA for 3D geometric descriptions originating from the gaming industry and is used for cross-platform geometric and kinematic model data. Other open source formats also exist, such as glTF – developed by the Khronos Group to be API-neutral (The Khronos Group Inc, n.d., n.d.).

Industrial engineering studies such as those from Åstrand & Rodahl (1986) and Kroemer, Kroemer & Kroemer-Elbert (2010) have extensive human body models tailored for industry. These models can be combined with game technology to guide the proper development of the digital models to include complex kinematics, such as fatigue and recovery curves – as proposed by Calzavara, Persona, Sgarbossa, *et al.* (2018).

Siemens's Jack software for their Tecnomatix platform is an example of well-developed, industrial applied kinematic modelling, along with fatigue and strain calculations. However, the software lacks the inherent real-time nature of game engines and individualisation of a DT.

### 4.2.3 **Psychological modelling**

Psychological modelling will enable action prediction, mental fatigue prediction, the ability to establish the attention and focus of a human and other beneficial information about his/her cognitive abilities and state when performing an activity. Research examples of psychological modelling for human centred systems include, among other models, the Belief-Desire-Intention (BDI), SAMPLE, and PECS models – as discussed by Elkosantini (2015). These models present their own model for human behaviour, taking psychological background, social background, and physical capacity into account, based on previous research done in each field. The *Endel Tuving Model of memory*, the *Atkinson-Shiffrin* model, the *Skills, Rules and Knowledge* model, as well as the *Abstraction Hierarchies* model are just some of the existing models on how humans process information and make decisions in different situations (Hedge, 2013).

### 4.2.4 **Biological modelling**

Biological modelling considers the human from any biological laws and behaviour. Behaviour of the cardio-vascular system, the effects of drugs, chemicals, atmospheric composition, illness, and other aspects and how it affects the kinematic and psychological models forms part of this model.

# 5 The 3SAL Activity Life-cycle for I4.0 communication federation

The I4.0 promises regarding horizontal and vertical enterprise integration require a common structure and ontology for communication between the cyber and physical worlds. Considering the role of the HRH-AS to delegate on behalf of the human, research was conducted on how to effectively communicate activity-related information. This section thus presents a common definition and structure of an *activity* to promote standardization of communication within the I4.0 and CPS context and thus facilitate the AL of managing the activities holons perform and the data associated with them. The content of this chapter was published and presented at the International Workshop on Service Oriented, Holonic and Multi-Agent Manufacturing in 2019 (Sparrow, Kruger & Basson, 2019b).

## 5.1 Communicating activity information

The dissertation defines an activity simply as a collection of actions, performed by an executor, which should lead to some world state. This definition infers that activities can be broken down into actions, which themselves can be broken into smaller sub-actions. The degree to which an activity and the related actions should be broken down – taking into consideration the implications for the communication of activity information – should thus be determined.

The sub-division of activities and their individual actions should be governed by consideration of the intended executor. The smallest explicit action an executor can be instructed to perform depends on the way they process information and the time it takes to process feedback from an action. For comparison, the smallest action that can be given to a CNC machine may be one line of G-code to execute; anything smaller, such as the switching of the transistors in its drivers, does not make sense to it. Similarly, when humans are given an instruction, they can only subdivide it to the limit of the thoughts reaching the basal ganglia[3], where the excitations of neurons to the muscles is outside of conscious scope and the limit of their reaction time. This gives a lower limit on the size of a describable action for a human executor.

In fact, the detail to which actions need to be defined is dependent on the flexibility of the process and executor, as well as the intelligence (and experience) of the executor. A more flexible process can afford an action plan that is less pre-defined. Similarly, when the executor is more intelligent and flexible, a less refined action plan would be required - the executor can find their own path to the next

[3] The basal ganglia are a group of structures at the base of the brain involved in action selection and motor plan to inhibit or facilitate movement.

action in the activity. Where a simple CNC machine that must paint a part would need an action plan that is refined (on its behalf) to individual movements, a human worker could be given the instruction to paint the part and the conditions of a successful paint job.

Furthermore, the collection of actions that constitute an activity does not always have to be the same. An activity is created to meet some goal, and the action path chosen to meet the goal can depend on the state of other world variables. Figure 13 demonstrates a simple grouping of similar action sets seen as the same activity. While some differences may exist in the nature and sequence of the individual actions, these sets of actions all aim to achieve the same goal.

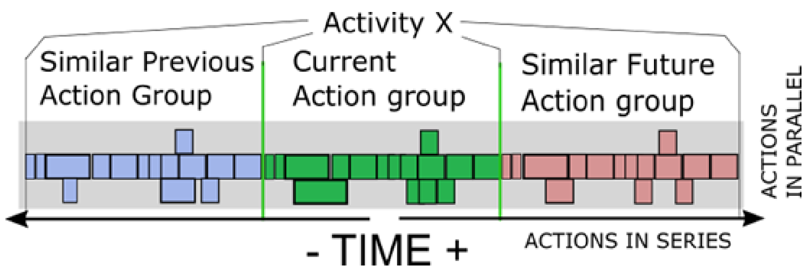



**Figure 13 – Grouped actions with similar goals are identified as the same activity type**

For intelligent executors, over-defining the action path for an activity destroys the flexibility and robustness of the executor, who would have adapted their actions to meet the requirements in changing circumstances. It is this ability to perform an activity with a minimally defined set of actions that makes a human worker such a robust and flexible resource. Contrastingly, the ability for machines to repeat a well-defined activity with high precision and speed sits at the core of why machines are valuable.

## 5.2 Activities and time

All activities are subject to time, as an independent variable in their execution. As such, the gathering, structuring, and communication of activity information is subject to time as well.

Observing the passage of time, there exists a bisecting line between future and past – the immediate present line. This line has no width (in the time dimension) and, not considering the effects of relativity, is the instant in time in which all matter, process states, and information exists in the moment between “about to” and “was just”. The problem with the word “now”, apart from mainly being used as an adverb and not a noun, is that it can also represent a spacious present and not the rigorous bisector between future and past. This bisector is defined here as the *origo*, from the pragmatic use of the word (Grenoble, 1998). There exists three deictic dimensions, each with its own origo (orientation point), of which the default is time.

Section 5.1 defined an activity as a set of actions. While these individual actions are completed as they cross the origo to the past, an entire activity is not completed until its final constituting action crosses the origo. This consideration of activities and their passage through time leads to the following important observations:

- Activities need to be scheduled before they are executed; therefore, existing entirely ahead of the origo.
- Activities will have an actual starting time that may differ from the scheduled time - marked as the point where the first action of that activity crosses the origo.
- Activities are executed to achieve a specific world state, and until the activity has failed or succeeded, at least one of its constituting actions is still intersecting the origo.
- Activities will end at some point in time. This occurs when the last action of that activity has crossed the origo to the past.
- Information on how the activity is scheduled, executed, and what the after-effects of its execution was, will be of value to future activities, as well as parties involved with (or affected by) the activity.

The above-mentioned observations indicate that the description of an activity would be affected by its relative position to the origo – either ahead (scheduled in the future), crossing (currently being executed), or behind (already completed in the past). These observations are used to create a structure for communicating activity information, as described in Section 5.3.

## 5.3 The three-stage activity life-cycle structure

From the discussion in Sections 5.1 and 5.2, an activity's life cycle can be marked by three temporal domains – henceforth referred to as the Three-Stage Activity Lifecycle (3SAL). The 3SAL implies that activities can exist in three stages: *scheduled*; *in execution*; and *completed*.

The activity is first *scheduled* – a specific activity type is instantiated along with a scheduled start time. At this stage, parameters and data unique to that activity instance can be set. A scheduled activity can still be removed, since it did not yet have a physical effect on the world, tools, materials, or the executor.

When the first action of the activity is carried out by the executor, the activity enters the *in execution* stage and now influences the physical world. This influence cannot be erased – even if the activity is cancelled, it still enters the *completed* state and information related to its effect on the physical world is recorded. The information of a cancelled activity may still be important, considering that the information could be related to e.g. tool wear, material consumption, executor state, etc.

When the final action (and thus the activity) is completed, the activity enters the *completed* stage. The activity then remains in a frozen state, with all the recorded information specific to that activity instance during its scheduling and execution phases. It is assumed that the activity will have a causal effect on the world even after its completion - whether the effects are detected immediately (e.g. through quality inspections) or only at a later stage (e.g. from an obtained customer review). The 3SAL allows data to be collected on the activity indefinitely in its third stage.

Along with the physical structure 3SAL offers, the three stages of an activity is colour coded to facilitate human readability when developing, or communicating activities. In the remainder of the dissertation, the three temporal stages of the 3SAL are colour coded to support readability and understanding ("Symbolism of Color: Using Color for Meaning", n.d.) – as illustrated in Figure 14. Activities in the scheduled stage are indicated in red, activities currently being executed are indicated in green, and completed activities are shown in blue.


**Figure 14 – The Three-Stage Activity Life-cycle**

## 5.4 Level of detail approaching the origo

There is a process of action refinement in activity planning whereby an intelligent executor, such as a human, will refine their action plan based on the constant gain of knowledge leading up to the point of taking action. Figure 15 illustrates this concept and shows how more detail is added to the action plan the closer the activity is to the origo. This refinement process continues to happen during the execution stage as well, and how fast an action plan can be refined, adapted and calculated relies on sensor placement, model accuracy, and processing power; however, the relationships between these aspects are not explored further in this dissertation.

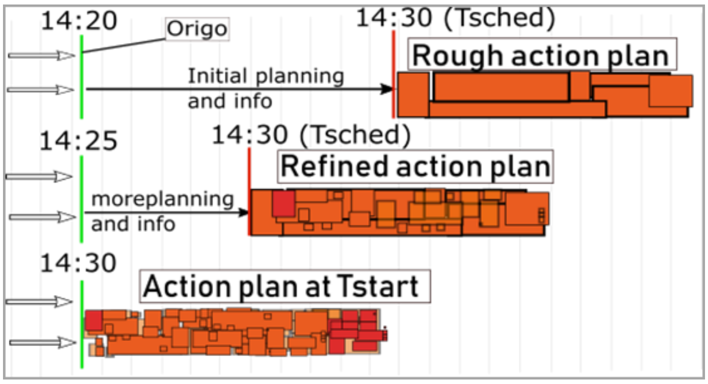



**Figure 15 – Action plan refinement for an activity approaching the origo**

## 5.5 3SAL activity data

Since activities are a known grouping of actions, they can be referred to simply by an ID to identify their type and unique instance. The 3SAL structure aims to standardize CPPS communication for scheduling, execution, and post execution data of activities. These efforts are to facilitate plug and work concepts (Monostori et al., 2016).

To facilitate the effective communication of activity information an activity data structure, referred to as an *activity data shell*, was developed. An activity data shell is a unique, lightweight and extensible container for activity instance information, which supports the storage of activity information throughout the three stages of the 3SAL.

The structure of the activity data shell, in each of the three stages of the 3SAL, is illustrated in Figure 16. The shell of a stage 1 activity consists of its ID or type, and its scheduled start time. For a stage 2 activity the actual start time is added, and for stage 3 the actual end time is added.

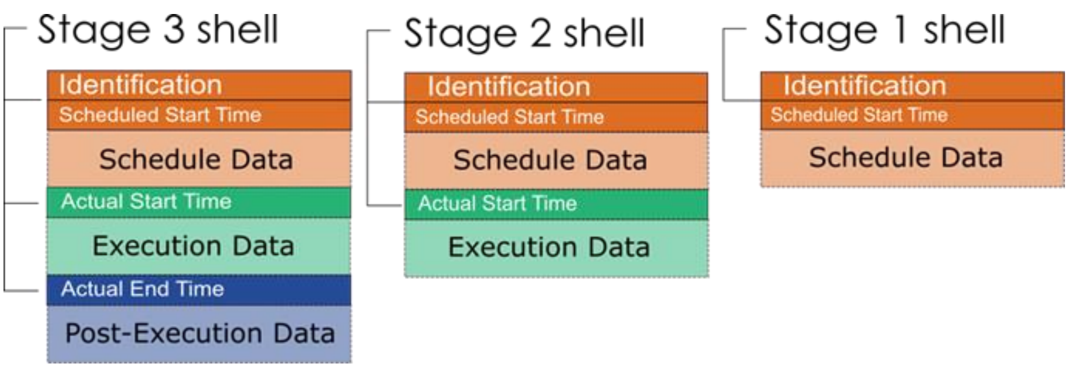



**Figure 16 – Shells and data blocks of stage 1, 2, and 3 activities**

An executor that is familiar with an activity type only requires the ID and a scheduled start time (i.e. the data shell of a stage 1 activity) to proceed with an activity instance. Fast, lightweight communication is advantageous in an I4.0 environment where a multitude of entities are constantly exchanging data.

Shells can be communicated before more detailed data allowing lower bandwidth communication and more efficient collaboration. The shells represent the most abstract description of an activity before more information is added to refine it.

While multiple instances of the same activity type can be generated, every activity instance will be unique - as caused by, for example, differences in duration, quality, action plan, etc. The activity data shell can be used to identify individual activity instances and store the data that uniquely describes each activity instance. The data that is recorded in each of the three activity stages is stored in three dedicated activity *data blocks*: Schedule Data, Execution Data and Post-Execution Data.

Schedule Data comprises all data that is available prior to activity execution, which may include Program Evaluation Review Technique data, deadlines, pessimistic and optimistic finishing times, customisations, etc. Data gathered or generated during the execution phase of an activity is contained in the Execution Data block. This data may refer to tool usage, action completion times, and process anomalies. The data that is gathered after the activity is completed, such as customer reviews, embedded sensor data, quality reports, or any information that becomes available after completion of the activity, is contained in the Post-Execution Data block.

It may be possible that data relating to earlier stages is only discovered during the later stages of an activity's life. For example, some assumptions made during the schedule stage is only confirmed or denied upon execution in which case the discovery is written to the execution data block, or a quality assurance test, or customer evaluation is only performed after the execution stage and is therefore written to the post execution data block even if it was related to an action during execution. 3SAL states data needs to be recorded to the data block of the stage it was received since valuable insight can be gathered from having the correct time perspective on information.

## 5.6 Value of the 3SAL to industry 4.0 environments

The 3SAL activity structure was created to federate the communication of activities and responsibilities between entities in a modern manufacturing environment. Using a common data structure, and generalising the identification of activities based on universal principles, will benefit the effectiveness of AL involved. The following sections discuss these benefits in more detail.

### 5.6.1 Reactive action plan refinement during scheduling

All knowledge of an activity that is yet to reach the origo (i.e. an activity that is still in the scheduled stage) relies on prediction, i.e. using some model, simulated inputs and the computation of resulting outputs. This prediction requires a balance between two restricting factors:

- It should be computed as late as possible (i.e. as close to the origo as possible) to ensure accuracy. The outputs of the computation will be more accurate if the variables used during computation will not change significantly by the time of execution, which implies that the accuracy of the prediction will deteriorate over time.
- It should be computed as early as is needed to allow for enough time to compute the model and integrate the resulting inputs in the schedule. The time required for computation will increase with model complexity.

The accuracy of models can be improved over time, as historical data is gathered and analysed. However, this process may be slow and lack the agility to adapt to rapid changes in the schedule (caused by disturbances or opportunities). The 3SAL structure can allow for more reactive model and action plan refinement in Industry 4.0 environments. Figure 17 shows how an executor closer to the origo with a more refined action plan (Executor B), can help an executor with a similar activity later in time (Executor A) to refine its action plan.

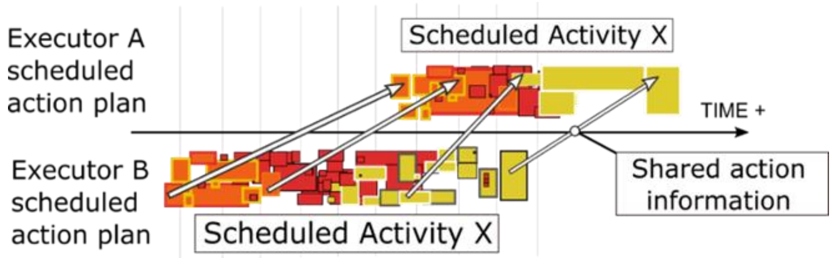



**Figure 17 – Reactive action plan refinement approaching the origo**

### 5.6.2 Activity model refinement through analysis

The level of process knowledge that can be obtained from a completed activity is dependent on the continued data acquisition and analysis of the residual effects of that activity. During execution of an activity, several factors may limit the amount of data that can be collected to improve the action plan. These may be sensor limitations, actions being too complex to analyse before the activity is completed, or certain information not being available yet (e.g. customer satisfaction).

If an activity is repeated or similar activities will be performed, it will be beneficial to improve the models describing the activity based on data gathered during execution and post-execution, so the action plans of future activities can be refined earlier. Analysing the existing activity models and predicted action plans and comparing them to the data obtained from execution and post execution, future activities can be refined or kept up to date for improved predictions and schedules.

To illustrate this Figure 18 shows data gathered during a later activity with process steps or data represented as blocks (green blocks gathered during execution and blue gathered post execution). The existing activity model is shown in red as the expected standard of actions or data. The current activity and the current model is then compared and improvements on the model is shown as an update in the data or process steps blocks.

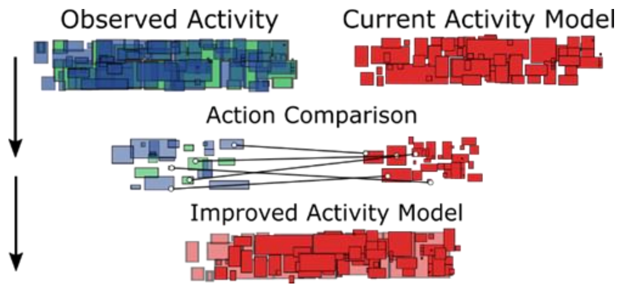



**Figure 18 – Improved stage 1 activity data from the data gathered in stage 2 and stage 3 activities**

### 5.6.3 Model Improvement Through Peer-to-peer Analysis

Any new cyber-physical executor plugged into a production system could inherit a copy of the current activity model, which would contain knowledge gained from the production environment, before refining it through time to match its own capabilities and methods. When different executors have performed the same activities, model improvement and optimisation is possible through peer-to-peer analysis of the performed activities.

The 3SAL structure of activity data allows for similar activities to be effectively compared within the temporal frames they exist. Actions and data from activities can be located faster, and with reduced memory bandwidth, when large numbers of activities must be compared for actions or data. This means that if all activities that are scheduled to work with a specific resource need to be found, executors could simply be asked for the stage 1 shells of those activities.

Figure 19 shows the comparison of two activities during stage 2 and stage 3. The balance between investing in sensors, models, and analysis has also been explored, along with the phenomenon that arises with the feedback of information across the origo, but it is outside the scope of this dissertation.

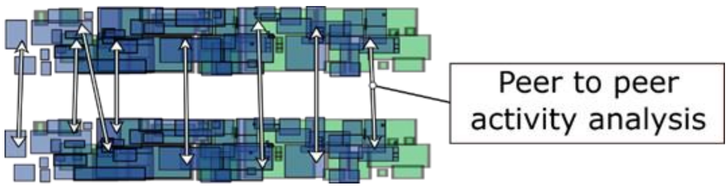



**Figure 19 – Activities can be compared between executors for improved analysis and activity model refinement.**

### 5.6.4 Improved Cooperative Execution Through Action Plan Refinement

Through repeated attempts of an activity or by training a human executor automatically builds and improves models of the activity and an action path for its completion, increasing success and decreasing time between actions in the lack of uncertainty.

If the human needs to cooperate with other digital entities in a manufacturing environment – Pacaux-Lemoine *et al.* (2017) showed that their model of a cooperative agent required the internal ability to build a model of the agent it was co-operating with. This allows it to predict its future actions and build its own action plan accordingly.

When two executors in a manufacturing environment need to collaborate on activities, synchronising actions and information becomes critical to successful cooperation. Knowing whether one executor has initiated an activity, how far they have progressed with the activity, or even what action they are busy with, will enable another executor to refine an action plan for its own execution. Figure 20 illustrates action plan refinement during execution from data before and after the origo.

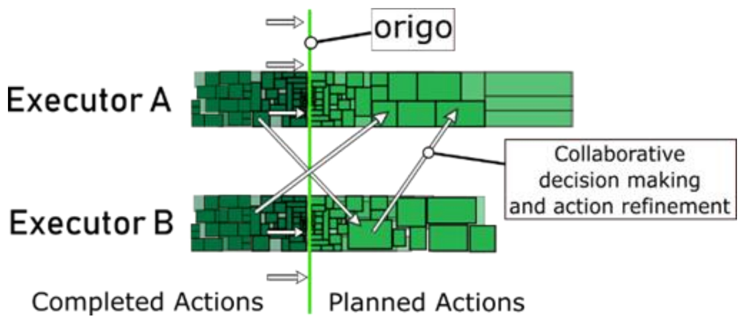



**Figure 20 – Cooperative execution and action plan refinement**

In the research by Pacaux-Lemoine *et al.* (2017), the model of a cooperative agent required the internal ability to build a model of the agent it was cooperating with, deduce its intentions, and refine its plan of action. Building the model from third stage activities, and using the origo to orient the synchronization of the co-operative activities, allows each agent to accomplish this. The model also equips an agent with the ability to communicate its current activity state and action plan with cooperating agents.

### 5.6.5 **Facilitating Administrative Logistics of Responsibility**

One of the four influences on organisational entities discussed in Section 3 is that of authority, or the ability to assign responsibilities. When a person is tasked with completing an activity, they are carrying responsibility assigned to them. Early in the study, it became evident that a generalised means of communicating activities, and therefore responsibility, was required – leading to the creation of the 3SAL activity model.

# 6 The BASE architecture

A reference architecture is an abstraction of design elements that is used as a guide for a specific implementation of the architecture. The Biography-Attributes-Schedule-Execution (BASE) architecture presented in this chapter aims to create the administration shell that, when combined with a human worker, forms a CPS that fulfils all the requirements of a Resource Holon – effectively forming a Human Resource Holon. To accomplish this, the BASE architecture is designed to facilitate effective AL between a human and other digital systems.

This chapter starts by introducing the principles that guided the development of the BASE architecture, before describing the core components of the architecture. Thereafter, the possible context-specific plugins to the architecture are discussed.

## 6.1 Guiding principles for the administration shell architecture

The architecture was created by the distillation and abstraction of a pin board of concepts and ideas. These concepts included needs reported from industry and literature, inspiration from existing ideas and architectures (e.g. from the online gaming domain), and describing the relationship between these concepts. The grouping and linking of items on the pin board attempted to address the laws of the artificial with scalability, stable design choices and providing a single source of truth of the HRH – in accordance with Valckenaers & Van Brussel (2016).

The architecture addresses the difference between Administrative Cognition and Administrative Logistics, by placing all decision-making functionality in designated plugin components and all data that flows between decision making components in generalised core components.

With the I4.0 vision in mind, a set of guiding principles for the development of the architecture was selected to support scalability, adaptability, and affordability. These guiding principles are listed below:

- **Market and vendor agnostic** with no prescription of solutions nor technologies. No ties to a specific manufacturer or developer nor specific technology to function.
- **Modularity** ensuring incremental and concurrent development, as well as facilitating upgradability and interchangeability of components.
- **Separation of concerns and capabilities** is a design principle that promotes modularity by encapsulating a set of information with a corresponding set of functions, thereby minimising the dependency of data from one concern to another.

- **Interoperability and integrability** which allows horizontal and vertical integration of different technologies in, or with, the administration shell.
- **Personalisation and optimisation** allow the idiosyncrasies of the operators, as well as the differences in environments and context, to aid in decision-making and operations.

Ruppert *et al.* (2018) set out guiding design principles for Operator 4.0 solutions, which coincide with the principles and visions presented above. One important principle is that of corporate and social responsibility. This principle is one of the emerging benefits of using the architecture presented here and is discussed in the evaluation in Chapter 10. The basis for communicating, handling, and processing activities is the 3SAL activity structure, as described in Chapter 5, which facilitates communication of activities between CPSs.

## 6.2 BASE architecture core components

The core components, or BASE, of the architecture is shown in Figure 21. The components of the BASE architecture are defined as Biography, Attributes, Schedule, and Execution. These components generalise and manage AL of the Holon.

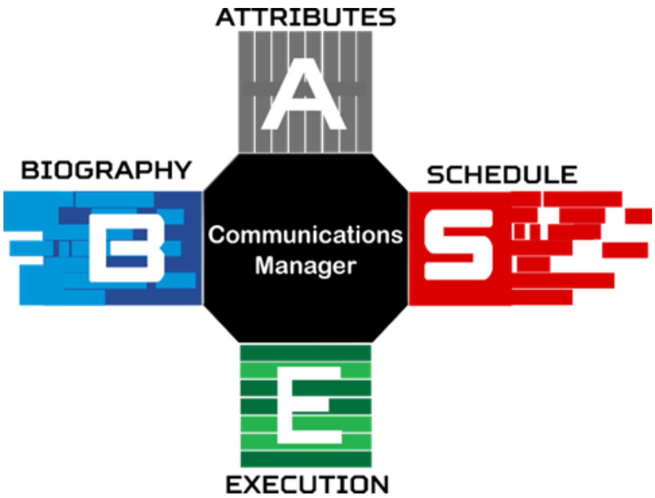



**Figure 21 – BASE architecture core**

The four components can be described as: Schedule – dealing with the future; Execution – dealing with the present; Biography – to deal with the past; and, overarching them all, Attributes – dealing with the properties of the HRH that are stable in time and context. The Communications Manager facilitates the communication between these components, and between the HRH and other digital systems. The following sections describe the role and functions of each of the BASE components.

### 6.2.1 Schedule

The HRH needs to keep track of planned activities and events with digital accuracy, as well as be able to communicate this information digitally without requiring the human's attention, time. The Schedule addresses the AL involved with planning and organising activities to be executed.

Schedule is a data-repository of the HRH that augments the human's ability to remember and communicate activities to be dealt with in the future. The Schedule is shown in a warm palette to indicate the nature of the activities it contains as described by the 3SAL structure. Activities can be added, moved and removed from the Schedule and the Schedule itself does not dictate any rules for these actions and has no intelligence of its own. An implementation of the Schedule consists of parallel running queues that contain pending activities and an Execution Gate that sits at the instantaneous present line (or origo, as described by the 3SAL).

The Execution Gate serves only as an orientation point between the scheduled time of activities and the origo. Only stage 1 activities can exist in the Schedule (as described by the 3SAL).

The Schedule is not limited to the human's activities - it also holds activities intended for the administration shell. Examples would be database maintenance, scheduled analysis activities or safety monitoring activities. This provides a unified view of the full HRH schedule. Figure 22 illustrates a Schedule instance.

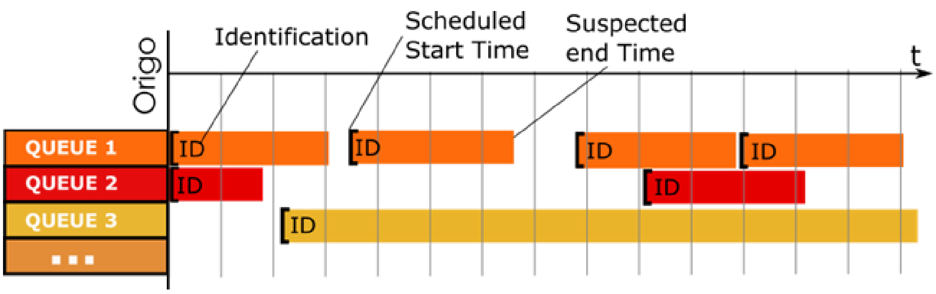



**Figure 22 – An example schematic of a Schedule instance**

A vast amount of research has been done on scheduling in different organisational structures - including holonic systems (Paulo *et al.*, 2007; Leitão & Restivo, 2006) - and how the Schedule is managed is up to the application and the context. Resolution of scheduling conflict is the responsibility of components discussed in Section 6.3.

### 6.2.2 **Biography**

The HRH needs to recall past events and activities with numerical precision and be able to communicate this data to other digital systems - the Biography facilitates the AL related to this data.

Biography comes from ancient Greek, meaning "life" (*bios*) and "writing" (*graphie*). Biography was chosen for its use on the life events of an intelligent entity. The Biography contains data of all completed activities of the HRH (stage 3 activities, as described by the 3SAL), as well as any relevant events, incidents or accidents the HRH was affected by or involved with. Figure 23 illustrates a Biography instance.

The HRH-AS can learn from analysing activities in the Biography to update and improve the Attributes discussed in the next section. This analysis would be performed by components discussed in Section 6.3.

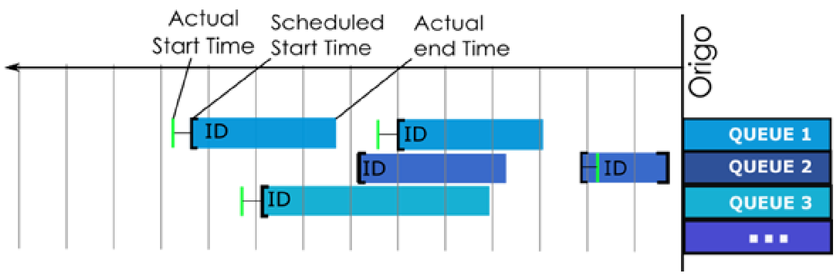



**Figure 23 – Schematic of a Biography**

### 6.2.3 **Attributes**

The HRH needs to maintain numerically detailed information on the physical, mental, and biological attributes of the human. Data about the specialisation of responsibilities the human has in the organisation and relationships with other entities are also required in various decisions. Details about the administration shell, such as versioning, user settings, and hardware and software information also form part of this data. The HRH can then provide this information to other digital systems or use it for its own decision-making purposes. The Attributes component facilitates the AL concerning this slow-changing data that defines the HRH in its environment.

Attributes is a specialized data repository, like the Schedule and Biography, but concerns itself with slow-changing, steady and predictable data that describes the holon type and traits. Where a name or ID number will constitute a static, timeless attribute, age and body weight will be time dependent, changing attributes.

As an example: muscle mass barely changes from one activity to the next, but exhaustion does; therefore, muscle mass is an attribute, while exhaustion is rather a state variable. *Stamina* will be an attribute that the HRH can use to determine exhaustion in this case. The architecture does make specific provision for the amendments and updating of Attributes based on information in the Biography.

A more concrete distinction between an attribute and a state variable is that an attribute can be considered static during the typical lifespan of an activity. As such, the system that handles the planning of an activity should be able to rely on the attributes to make per-activity predictions.

The attributes of the HRH are classified in two types: Personal Attributes and Contextual Attributes. Persistent data about the human and his administration shell is referred to as Personal Attributes and forms a digital model of the human (as discussed in Section 4.2). Some examples of Personal Attributes would be biometrics, language, physiology, favourite colour, and the preferred font for displayed text. Other attributes of the HRH may be specific to an application context. These attributes are called Contextual Attributes and represent a model of what the HRH is to the context it finds itself in. Examples of Contextual Attributes are access permissions, job description, qualifications, or activity related attributes, such as Standard Work. Personal attributes describe the human as a biological individual, and the administration shell as a digital tool - regardless of the environment and context. In contrast, Contextual Attributes can change from one context to the next.

The difference between Personal and Contextual Attributes can be further illustrated through an example: a worker named Chell is 24 years old, weighs 76 kg and can complete an average of 35 assemblies a day for a specific product. Chell's name, age and weight hold true for any context, while her productivity can only be related to the specific operational context. As such, Chell's name and weight are Personal Attributes, while her productivity is a contextual attribute. Figure 24 illustrates some Personal and Contextual Attributes of a worker.

In terms of holonic design, Contextual Attributes define the specialisation of the HRH in the administration shell and provides a means for the communications of skillsets, permissions and abilities to other holons while Personal Attributes Allow for a more human aspect to play a role in the HRH-AS that is not strictly business oriented.

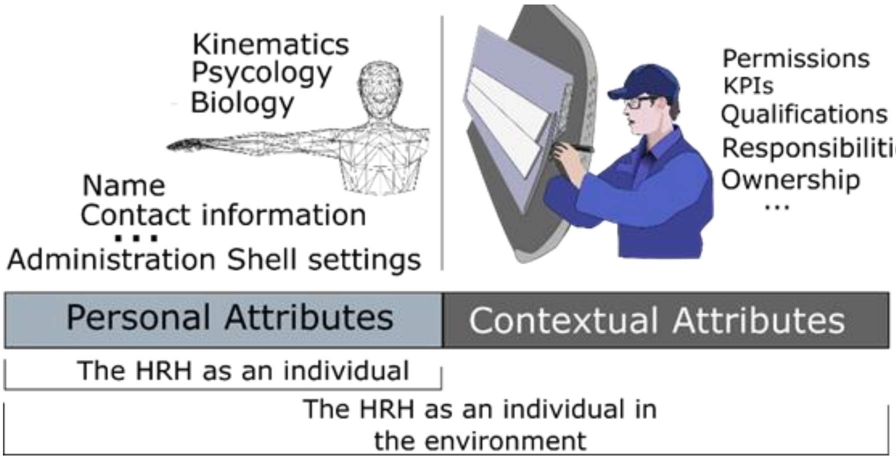



**Figure 24 – Personal and Contextual Attributes**

While attributes can be completely static and timeless (as is the case with an individual's name or identity number), they can be observed to have slow, steady and predictable changes in their values. For example, an individual's age will surely change slowly over time, and so may their body weight and productivity. The BASE architecture makes specific provision for the amendment and updating of the HRH Attributes over time, based on the information recorded in the Biography – this is discussed further in Section 6.3. This amendment and improvement of Attributes aim to ensure that the HRH is represented as accurately as possible, so that the data can be used to facilitate HRH scheduling and execution decisions.

Furthermore, Attributes allow for the personalisation of safety monitoring, workstations, health management, and other aspects that could vary between HRHs. As such, the administration shell of the HRH can use the data of the Attributes component to tailor itself to the needs of the human.

### 6.2.4 **Execution**

Execution deals with volatile real-world information about the HRH, referred to as State Variables (SVs), where Attributes dealt with persistent data. The separation of volatile from persistent data depends on the context and activities being performed; however, very short-lived data will generally be the responsibility of the Execution component.

The human's gaze direction, position, velocity, and intentions or spoken words are examples of SVs managed by Execution. The Execution component is arguably one of the most complex of the core BASE components.

It is also responsible for facilitating the AL involved with interfacing with the human, which can be described by two processes:

- Data flow from the human to the cyber world, by gathering and consolidating the state information, as well as information the human wishes to communicate explicitly to provide a single source of truth; and
- Data flow from the cyber world to the human, by translating and rendering digital information from the administration shell or other CPSs for the human to consume.

As such, Execution provides a flow of data from the human to the administration shell and other I4.0 components in the environment, which can then make intelligent decisions based on the human and his WOI. Execution also provides a means of communication to the human from external digital systems or internal BASE components.

### 6.2.5 **Communications Manager**

BASE components require communication between each other and external holons and digital systems. This communication requires the management of connections through networks, security, data integrity, robustness and possibly consolidating different communication protocols. It is redundant to require each BASE component to serve this purpose and thus the management of communication between internal and external components is the responsibility of the Communications Manager.

To external holons, the Communications Manager is the face of the HRH. The Communications Manager distributes incoming queries for the HRH to the appropriate inner components and back to the external connection, possibly translating and formatting the data to suit either end.

The scope of different information communication and storage requirements among I4.0 components are expected to be wide - ranging from small, low-latency streamed packets for monitoring, to large blocks of data related to process instructions or product information. Therefore, a single communication protocol may not be suitable for all communication needs in an I4.0 environment and the Communications Manager has the responsibility of consolidating these requirements.

## 6.3 Plugins to the BASE architecture

The core components of the BASE architecture are common to any application and, as such, the implementation of only the core components in a specific context will probably not add significant value to stakeholders. For a specific context or application, the BASE architecture must be extended, customized, and integrated with regards to the following functions:

- The scheduling and management of activities in the Schedule.
- The management of activity data and execution.
- The collection and management of post-execution data in the Biography.
- The analysis of biographic information to adjust Attributes, provide insight to stakeholders and support future decision-making.

This extension and customization of the BASE architecture is facilitated through plugins to the architecture. The plugins are integrated as the corners of the BASE architecture, as is depicted in Figure 25. The following sections will introduce the four groups of architecture plugins, namely: Scheduling Plugins; Execution Plugins; Reflection Plugins; and Analysis Plugins.

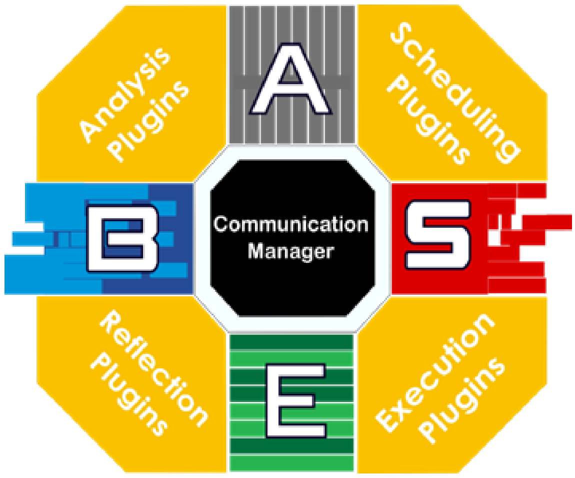



**Figure 25 – The full BASE architecture showing the core and plugin components**

### 6.3.1 Scheduling Plugins

The Scheduling Plugins (SPs) represent a set of tools, algorithms, software systems and decision-maker interfaces, which create, manage and optimise the scheduled activities of the HRH. stage 1 activities are placed in the schedule when the HRH accepts the responsibility to deliver a service to an external logistics holon, or the HRH-AS needs to perform some planned internal function.

As is shown in Figure 25, the SPs interface with the Communications Manager, Attributes and Schedule components of the BASE architecture. Through the interface with the Communications Manager components, the SPs can receive external or internal requests and notifications that relate to the scheduling of activities within the HRH. The SPs can then use the data in Attributes to optimise and personalise scheduling decisions for the HRH.

### 6.3.2 **Execution Plugins**

The Execution Plugins (EPs) are responsible for starting and managing the execution of scheduled activities. The EPs take stage 1 activities from the schedule and instantiates their execution graduating them to stage 2 activities by monitoring and communicating with the human through the Execution component.

EPs have access to the SVs in the Execution component, as well as the Attributes. EPs can make intelligent decisions based on this data on how to manage the activities the HRH needs to perform.

Where the SPs can provisionally accept an activity to the schedule, the EPs verify that the world is in a state that will allow the activity to complete. If something prevents the execution from proceeding, the SPs and EPs need to collaborate on how to handle the situation.

The EPs are also responsible for any virtual execution that is required as part of the resource holon responsibilities, where the Attributes of the HRH will be used to run a virtual activity execution.

### 6.3.3 **Reflection Plugins**

The Reflection Plugins (RPs) create and maintain biographic entries of completed activities or events. When an activity has been completed, it is promoted to a stage 3 activity by the RPs and it is entered into the Biography. Data about the events of an activity can still be gathered post execution, e.g. through reviews or quality checks.

RPs allow the addition of post execution information to activities in the Biography. Examples include quality checks, customer reviews, or any other data that can be used in correlation with the execution and scheduling data of an activity to improve processes or traceability. The RPs thus enable quality feedback to factory workers, which often never see the work they've done after it leaves their work station (Murthy, 2018). Feedback and analysis on their techniques can be referenced to specific activities, which will greatly improve their ability to learn from mistakes or confirm their decisions made in certain circumstances.

### 6.3.4 **Analysis Plugins**

The Analysis Plugins (APs) generate value from the data recorded in the Biography with the aim of updating the Attributes. The APs close the information flow loop of the BASE architecture by updating Attributes from Biography, which enables self-improvement, self-analysis and self-optimisation of the HRH. The cycle repeats with the SPs utilising the updated Attributes to make better scheduling decisions and the EPs better execution decisions.

The APs enable the generation of valuable information for process improvement, product improvements and insight. An example of a value-generating plugin would be the analysis, tracking and predicting of true Standard Work, which Murthy (2018) indicated was the most time-consuming part of implementing lean manufacturing processes on the shop floor. Currently, stopwatches are used to gather time data for work study methods and automation of this process, as facilitated by the BASE architecture, would be extremely valuable.

With increased traceability, workers could be allowed more flexibility in performing activities and trying new ideas. The results could be empirically evaluated from data stored in the Biography.

Multiple plugins from different stakeholders can be instantiated and concurrently executed. Unions could run monitoring plugins on behalf of the worker to ensure safe working practices are followed, tool manufacturers could run analyses for improvement of their products, companies could spot retraining requirements before major defects show up on products, or to adjust Contextual Attributes to improve a process.

## 6.4 BASE architecture discussion

The following sections discuss the design of the BASE architecture, the nature of information flow through its components and how it behaves in different contexts. Finally, the value of using the BASE architecture is discussed according to various qualitative measures from a developer and stakeholder point of view.

### 6.4.1 **Design**

The design of the BASE architecture followed the principle of separation of concerns, as a means to maximize convertibility, increase robustness and aid in development. This separation is achieved through two important mechanisms: the distinction between generic and context-specific components, functions and data; and the progression of activity information over time.

In terms of facilitating the management of AL, which integrate the human worker with other organisational entities, the BASE architecture separates the generic AL functions, common to all interacting entities, from the Administrative Cognition specific to the entity, in accordance with the principles recommended by Valckenaers & Van Brussel, (2016). Subsequently, all decision-making functionality is to be housed within the context-specific plugins. Figure 26 shows this distinction between the core and plugin components.

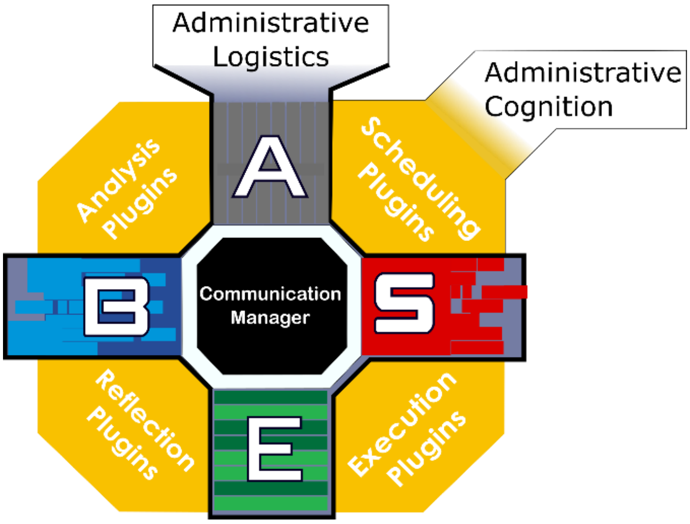



**Figure 26 – BASE separating generalised Administrative Logistics and context-specific Administrative Cognition.**

The separation of concerns is further guided by the universal and fundamental property of time – both in terms of the progression of time, and the changes in activity information that occur over time. These properties are encapsulated within the 3SAL model, which supports the progression of activities over time (seen as horizontal separation in Figure 27). The BASE architecture also acknowledges that data can be static or volatile with relation to time - volatile State Variables and execution information in the Execution component are thus separated from stable data in the Attributes component (seen as the vertical separation in Figure 27).

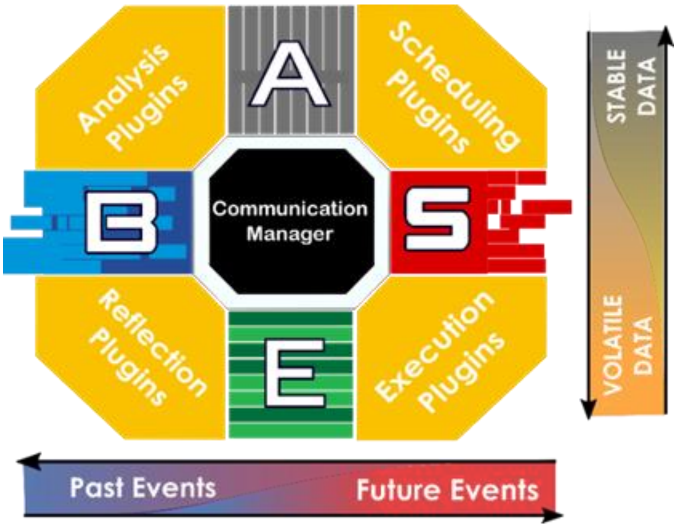



**Figure 27 – Separation of universal information properties reflected in the BASE architecture**

Since each core component's management is the responsibility of one adjacent plugin component, all plugin components share the same separation of universal concerns that the data in the core components have. The separation of decision-making components from shared data allows for these active and value-adding components to remain modular, while ensuring the integrity of the data they need to access.

### 6.4.2 BASE In relation to Human Cyber-Physical Systems

Humans have an exceedingly large variety of roles they can play in an organisation. All business processes are defined by a combination of decisions and actions, and humans take part in both extensively. A supervisor will perform tactical activities such as monitoring while a machinist will perform more operational activities making parts. The Attributes component describes the different capabilities and roles a human can perform in their environment, while their Schedule allows for optimal use of their time between assets that need them. The human's current state being present in the Execution component allow real time decision making with these assets. The human's Biography allows analysis and traceability on decisions, actions, and events that increases process and shop floor visibility.

These properties are not only useful when a machine or other humans require the right person for a task, but helps the human communicate this information to find and communicate with a machine they may want to interact with. BASE therefore enables a symbiotic mutualism and not just commensalism. Management not only has access to machine logs for factory optimisation and planning, but can draw from the information held by human BASE administration shells.

Internal processes and the relationships of authority, control, and ability change as company structure, products, or unforeseen events happen. When the communication channels and logistic structures between humans and other factory assets cannot match the new requirements with agility, there will be a separation between the static digital or machine infrastructure and dynamic human relationships and roles. This removes humans from the decision-making loops in the digital environment, reducing effective balance of authority, control, and ability between humans and machines. A BASE administration shell facilitates in this by making the HRH available to factory assets and decision makers without requiring the physical presence of the human. Other company assets can re-arrange and structure themselves around the human's state and Attributes. Information flow to and from the human is improved, and repetitive tasks such as answering questions about skills, schedule, current activities or historic information is automated so the human does not need to divide their attention.

In terms of the three principles Pacaux-Lemoine *et al.* (2017) discusses:

- The human is always aware of the situation since his BASE HRH-AS provides a universal point of contact for other machines and humans, and can retrieve data from other assets the human needs in a similar manner.
- Repetitive decisions and tasks such as answering questions about schedule, skills, current tasks or historic information on tasks can be automated.
- Mental workload can be carefully regulated since the state information, current task, and skillset/abilities of a worker can now form part of the decision making processes of machines around them.

### 6.4.3 BASE HRH-AS in a holonic manufacturing environment

According to Valckenaers (2018), humans require augmentation to participate in holonic manufacturing environments. The BASE architecture provides this augmentation in the form of a digital administration shell that elevates the human worker to a Human Cyber-Physical System as a Resource Holon. An HRH-AS implementation of the BASE architecture will act as the administrator in the cyber realm, elevating the human worker to a HCPS. Figure 28 illustrates the BASE administration shell connecting the human to a holarchy of other resource and activity holons. This figure shows that a BASE administration shell could be used for other resource and activity holons, but these resources could also be administered by other holonic architectures that allow the same AL protocols.

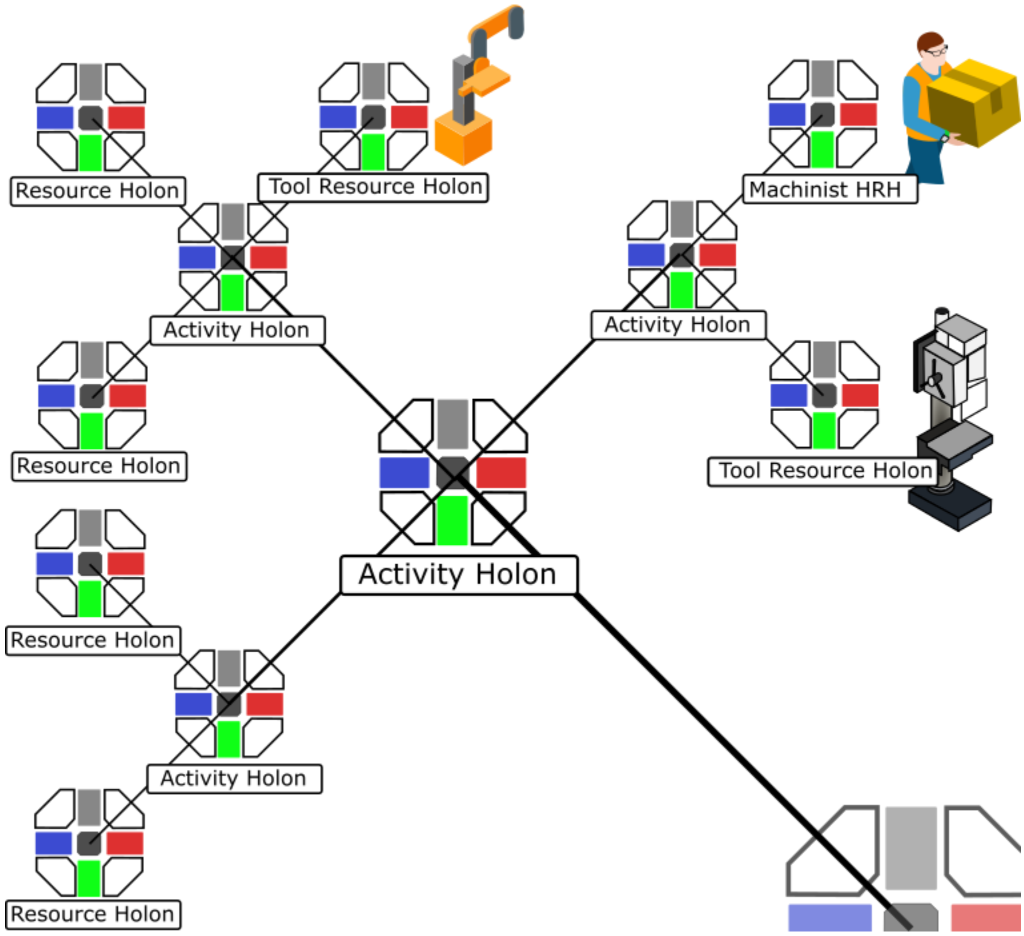



**Figure 28 – BASE administration shells in a holarchy**

### 6.4.4 The BASE architecture and the Human Digital Twin

As mentioned in Section 2.2.2, a distinction is made by Kritzinger et al. (2018) between a Digital Model, Digital Shadow and a Digital Twin. The BASE architecture's components can be masked to show the existence of these three concepts inherent to the architecture's design.

A Digital Model of the HRH exists when only considering the Attributes – they represent a static model of the HRH as model parameters for simulation on the HRH in its current form. The Biography, RPs and APs, along with the Attributes, form a Digital Shadow of the HRH by allowing for the automatic updating of the Attributes.

With the SPs, Schedule, EPs and Execution, an information loop is completed, and a full Digital Twin exists within the BASE architecture. The BASE architecture is not solely a Digital Twin, but the Digital Twin emerges as an integral part of the inner workings of the architecture. Figure 29 illustrates the emergence of these three concepts.

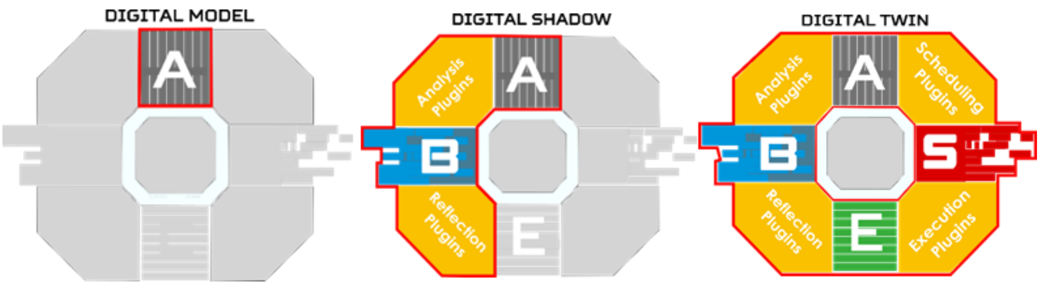



**Figure 29 – The BASE architecture components as a Digital Model, Digital Shadow, and Digital Twin**

### 6.4.5 Supporting development

There are several expected benefits of using the BASE architecture for development and planning of a digital administration shell and accompanying plugins. These benefits are as follows:

- The BASE architecture was designed so that the core components are generic to any implementation of the architecture, allowing development in any language or platform.
- The distinctions that the BASE architecture makes between its components does not cause any ambiguity when choosing where functionality or data should reside. As such, developers working on plugins can have specific domain knowledge without worrying about inter-dependencies.
- The BASE architecture promotes modularity, which means incremental and concurrent development is possible.
- The separation of data components through time, and data components from decision-making components, facilitates Plug-and-Produce of active components without affecting other components.
- The modularity and separation of data facilitates upgradability, since there is a clear distinction between old, new, and currently used data.

### 6.4.6 The BASE architecture self-improvement loop

The BASE architecture has a natural flow of feedback from past events, through the APs, to update Attributes. This flow enables improvements in scheduling and execution. This structure promotes continuous and automatic quality improvement similar to the Plan-Do-Check-Act (PDCA) cycles, which are a key concept of the Toyota Production System and lean manufacturing (Rother, 2010). The PDCA cycle and BASE architecture information flow cycle is compared in Figure 30.

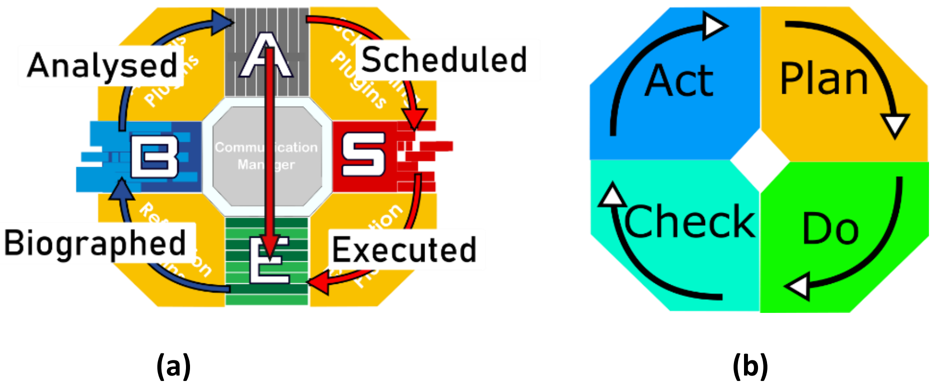



**Figure 30 – a) The flow of information in the BASE architecture and b) the PDCA cycle reproduced from Rother (2010).**

### 6.4.7 The BASE administration shell enables Plug-and-Produce.

The BASE Architecture enables a Plug-and-Produce capability described by Ulrich (2016). The core components describe the holon as an individual before the plugin components adapt it to a specific context, enabling the holon to change contexts along with its physical resource. The data related to the resource's identity and history is preserved and can be used in the new context. Figure 31 illustrates how one BASE core can be moved to a new context, with a different set of plugins to adapt its functionality.

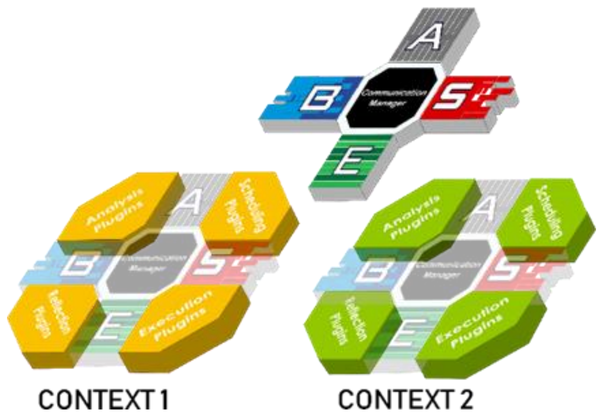



**Figure 31 – The BASE architecture core moving from one context to a next**

### 6.4.8 The BASE HRH-AS for improved Administrative Logistics

The HRH-AS aims to mitigate many of the shortcomings of AL when humans interact with machines, or other humans. The administration shell provides faster, more direct information channels, handles data storage and distribution with digital accuracy and speed, and can execute background administrative tasks without requiring the human worker's time or attention. Subsequently, workers would be able to spend more time on value-adding activities.

The BASE HRH-AS, and the technologies developed to support it, aim to provide a generic, effective AL layer to organisational entities. This will allow businesses to concentrate on making and changing business decisions without worrying about the cost of the AL to support them.

## 6.5 Summary

Chapter 7 discusses the implementation of the BASE architecture core – building on the description of the architecture presented in this chapter. The following summary offers a recapitulation of the roles of the various architecture components:

- To augment the human's ability to communicate and remember past activities and events with digital precision, the **Biography** serves as the data repository for stage 3 Activities and events.
- The **Analysis Plugins** serve to generate value from the data stored in the Biography and update the Attributes to enable self-adaptation and optimisation.
- **Attributes** augments the human's ability to remember and communicate personal and contextual data. Attributes also stores persistent data about the administration shell, providing a single source of truth for stable data on the HRH.
- The human's ability to remember and communicate upcoming tasks and events with digital precision is augmented by the **Schedule**, where the **Schedule Plugins** generate value by the management and creation of stage 1 Activities in the HRH Schedule.
- To direct the human as the physical resource of the HRH, the **Execution Plugins** generate value by utilising the Attributes and the State Variables from the Execution component to intelligently progress the execution of scheduled activities.
- The **Execution** component provides the capability of communicating digitised variables on the human and WOI state, as well as provides the ability to communicate to the human through available interfaces.
- The **Reflection Plugins** generate value by the creation and management of stage 3 Activities in the Biography, providing the Analysis Plugins with succinct and organized data to work with.

# 7 Implementation of a BASE Administration Shell

In Chapter 6, the core of the BASE architecture was proposed to be generic to all applications. Therefore, the core components could be developed independently and then be adapted to specific applications using the plugin components. This chapter describes the implementation of the core components of the BASE architecture, while Chapter 9 discusses the development and implementation of plugins for specific case studies.

The description of the implementation will be approached from a top-down point of view – the overarching control and orchestration principles of holons with BASE administration shells will be discussed, as well as how these holons interact with each other. After the discussion of the orchestration principles, the internal structure and implementation of a BASE administration shell, and how the internal components of the BASE administration shell interact, will be described.

## 7.1 Implementation strategy, language and tools

The implementation of the BASE architecture administration shell needs to orchestrate the concurrent interaction between the different architectural components. The support for concurrent communication was a key consideration in the selection of an appropriate programming language for the implementation.

Erlang, a concurrent functional language that is supplemented with a set of robust libraries in OTP, was selected for implementing the core components of the BASE architecture administration shell. While other languages or frameworks were considered, such as a multi-agent system developed with the Java Agent Development (JADE) framework, the benefits offered by Erlang have been highlighted by Valckenaers & Van Brussel (2016), Kruger & Basson (2019), and Hawkridge *et al.* (2019).

The BASE architecture does not prescribe an implementation language, nor does it specify the method of storing data - different databases could be applicable for different implementations. For this case study, Erlang Term Storage (ETS) tables were considered to be sufficient. Erlang can be integrated with many other databases, such as Riak, CouchDB (implemented in Erlang), and graph-based databases, which should be considered if a commercial product is developed.

The BASE implementation followed the standard practice for Erlang programs to be structured according to a supervision tree, which can be seen in Appendix A. The implementation of supervision trees increases the robustness and reliability of the software.

## 7.2 Orchestrating services

A collection of holons that are arranged (either manually or automatically) to deliver a particular service is known as a holarchy. Holarchies are *fractal* or *recursive* in nature, i.e. the individual holons that comprise a holarchy may contain holarchies themselves. In a BASE architecture holarchy, a central Orchestrating Holon will form service delivery contracts with subordinate resource holons to deliver services it needs to complete its own responsibilities. An Orchestrating Holon can be a resource holon, like a worker needing the services of a tool, or an activity holon that brings together resources to fulfil a task. Figure 32 illustrates part of a holarchy as it may appear in a manufacturing environment and offers a visualisation of the fractal nature of holarchies.

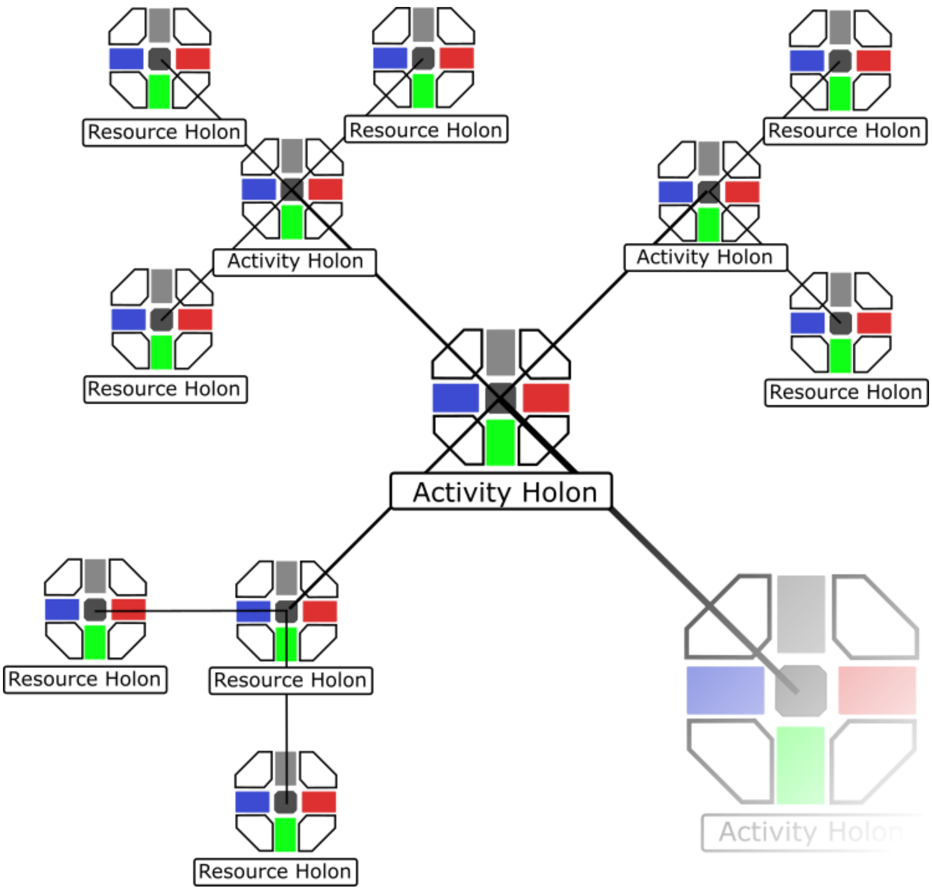



**Figure 32 – A visualisation of a possible BASE holarchy**

In Figure 32, each connection between two BASE administration shells represents an active BASE Architecture Service Provision Contract (BASE-SPC). The following section discusses how this BASE-SPC functions was implemented.

## 7.3 The BASE Architecture Service Provision Contract

When an Orchestrating Holon requires a service from another holon, it starts by finding the Business Cards (BCs) of holons that can provide the services it needs. The BCs provide details about holon capabilities and communication details to the orchestrating holon. The process of finding BCs and a detailed description of what they are is discussed in Section 7.4.

This section assumes that the orchestrating holon has obtained the BCs of relevant holons. There are six distinct steps in the interaction between an orchestrating and service-providing holon, as summarised in Table 1.

**Table 1 – BASE service provision conversation steps**

| Step | Description |
|---|---|
| 1 | The Orchestrating Holon sends the service-providing holon a Request For Proposal (RFP) that details the service it wants delivered and parameters or data associated with it. |
| 2 | The service-providing holon considers the details of the requested contract to determine if it can fulfil it. This process takes place in the SP of the holon, since it involves business decisions that rely on the holon's attributes and schedule (the two BASE core components that flank SP). |
| 3 | The service provision holon can either refuse the contract or send a proposal that may contain details of its lead time and expected quality of outcome. It could also request certain conditions be adapted before it can agree to the contract. |
| 4 | On receiving this proposal, the Orchestrating Holon can refuse the proposal, accept the proposal, or adjust its requirements and request another proposal. |
| 5 | On receiving an acceptance of the proposal, the service-providing holon then confirms the contract with a signature, and is officially a subordinate member of the Orchestrating Holon's holarchy. The service is delivered and reported on by communication related to the contract. |
| 6 | The SPC can be cancelled by either holon, or reach some condition of fulfilment, at which point the service-providing holon is released from the holarchy. |

Figure 33 illustrates the six steps listed in Table 1 between an Orchestrating Holon requiring the services of a Service Provision Holon. Solid arrows are calls initiated while dotted arrows are replies to those calls. Bi-directional arrows indicate the call can be made from either party.

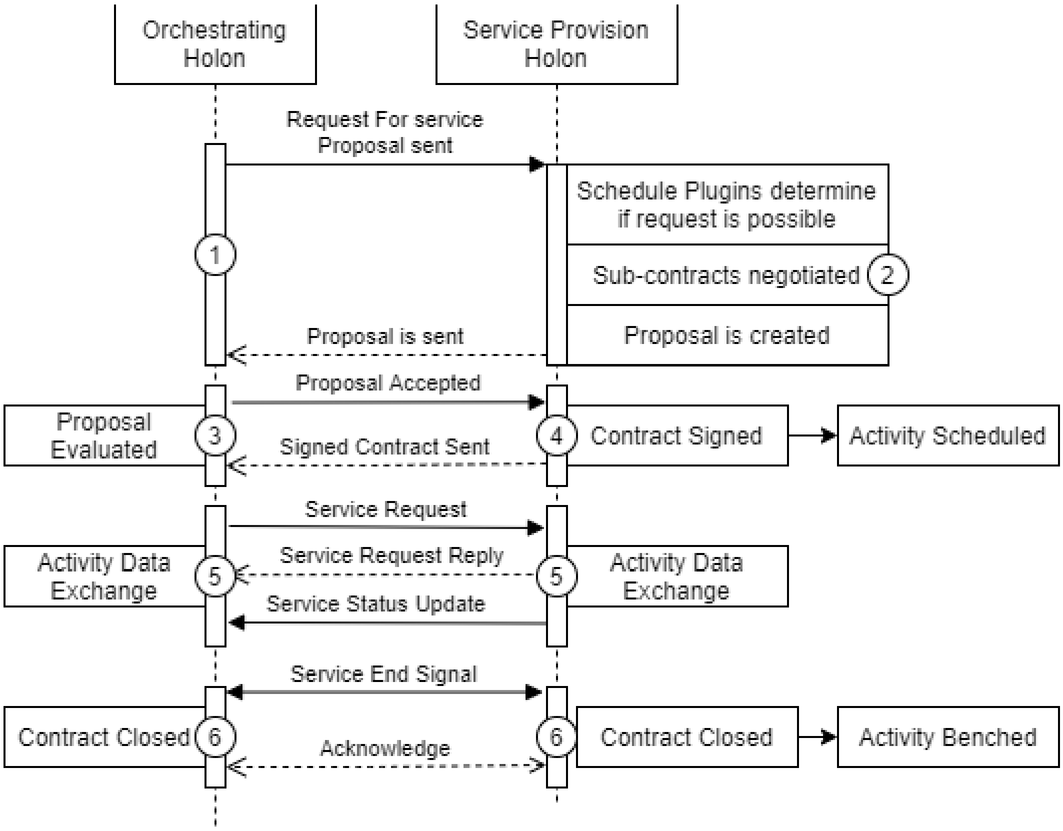



**Figure 33 – BASE-SPC conversation**

## 7.4 Resource discovery

To establish working relationships, holons need to be aware of each other, know how to contact and communicate with each other, and know what services they can request or provide. This information will be commonly required throughout the working life of a holarchy and can be generalized into standard form to simplify the AL related to this information.

The BASE Business Card (BASE BC) addresses this issue. A BASE BC is a small data structure that can be passed around between holons to share information about other holons in the holarchy. Table 2 details an implementation of this data structure. As an example, Figure 34 illustrates what this information may look like on a standard physical card if it was printed.

**Table 2 – BASE Business Card data structure**

| BC object | property | description |
|---|---|---|
| Identification | Name | The commonly used name for the person or thing that this BASE profile administers |
| | ID string | The business, country, or other identification string used to uniquely identify this holon |
| | Role/position | Role identifier or other prefix |
| Addresses | Erlang global address | The Erlang Process ID or atom used to identify this BASE communication manager component |
| | Ip v4 address | The IPv4 address of this BASE communication manager component |
| | Physical address | The room number, GPS coordinates or any other means of reaching the person or thing if it has physical form |
| | Email address | The email address of this person or possibly thing |
| Services | Service description 1 | A description of a service holon can provide |
| | Service description 2 | |
| Protocols | BASE Erlang protocols | Identifies the communication, data or other protocols that this holon understands |
| | BASE JSON protocols | |
| | BASE XML protocols | |

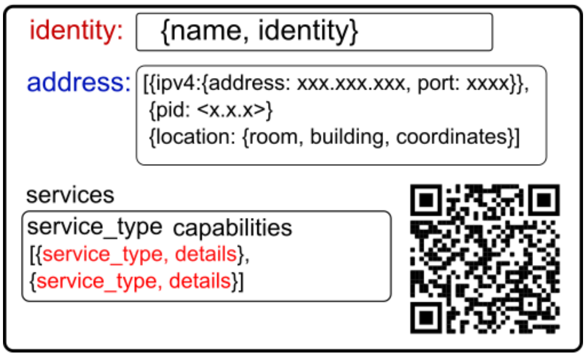



**Figure 34 – Illustration of a possible physical BASE business card**

BASE BCs can be passed between holons, or a dedicated service directory service can be used to provide holons with BCs for the services they request. Orchestrating holons that want to keep working with the same service provision holons for future activities can store their BASE BCs to directly contact them without having to perform a search on the network.

## 7.5 Internal structure of a BASE Administration Shell

A BASE administration shell is defined by two groups of components: the generic core components and the context-specific plugins. The BASE administration shell is implemented using the standard Erlang practice of supervisor processes that ensure that groups of other processes stay running, or are restarted if they crash. One top level supervisor oversees the starting of the five core component supervisors. Each of these supervisors are responsible for ensuring its component processes are running. Appendix A illustrates this supervisor tree for the core components.

The Biography, Attributes, and Schedule components are implemented in a very similar manner, serving as specialized data stores on which Create, Read, Update, and Delete (CRUD) operations can be performed. Schedule and Biography deals only with 3SAL activities, while Attributes serve as a more generalized database for any contextual or personal values that describe information that can be considered attributes. The Communications Manager performs several, more complex roles, including communication with external holons – discussed further in Section 7.6.

A BASE core component consists of three distinct permanent Erlang processes, namely *supervisor*, *reception* and *component* processes. The supervisor process spawns and maintains a reception process and the component processes under it. The reception process is responsible for communication with other core components, and filters messages to and from the component process. The reception process also handles basic queries about the component.

Figure 35 shows an illustration of this generalized core component structure. Using this generalized three-component template, the individual core components are described in the sections that follow.

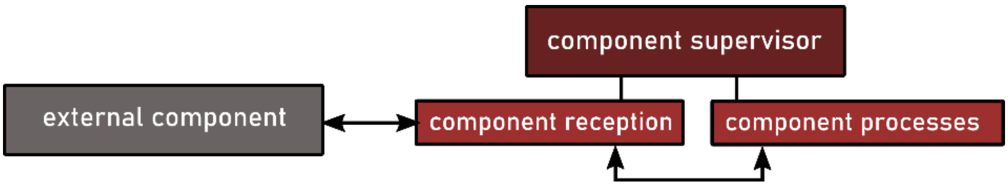



**Figure 35 – BASE core component generalised structure**

### 7.5.1 Attributes implementation

The reception process of the Attributes component receives and filters CRUD requests for the Attributes data store. It can check if the sender of the request is authorized, and ensure the message has the correct structure before committing it to the component process. The Attributes component processes manage the data store and backups of the Attributes.

### 7.5.2 Schedule and Biography implementation

The Schedule reception process receives any CRUD requests on stage 1 3SAL activities that is stored in an ETS table owned by the component process. The reception process ensures that only stage 1 activities are involved with the Schedule component. The Biography reception provides the same functionality as the Schedule reception, but filters for stage 3 3SAL activities.

### 7.5.3 Execution implementation

The Execution component serves a more complex role than the Schedule and Biography components when handling activities, being responsible for managing the data involved with their execution. The Execution component is also responsible for managing the digitized state of the holon. The Execution component is defined by three distinct internal components apart from its reception and supervisor processes.

The first of these three is the State Blackboard (SBB). The SBB is a short-term data store that holds the digitized representation of the holon. Examples of information that it may contain is position, temperature, current location or communicative data. The SBB serves as a synchronous, single source of truth on the HRH's current state. The SBB also keeps track of the WOI, as specified by the ARTI architecture. This ensures any critical data for execution, as well as safety monitoring, will be available to the components of the HRH-AS and can be communicated to external holons.

The second component is the Activity Handler (AH). An AH is a transient component spawned to aid in managing the AL of an executing activity. A stage 1 3SAL activity is handed to a newly spawned AH and, along with the appropriate EPs, the activity will transition to stage 3 where the AH hands the completed Activity to the final Execution component for temporary storage.

The final Execution component is the Execution Bench. This is implemented as an ETS table that acts as a holding area for stage 3 activities waiting to be biographed. Just as it is the responsibility of specific EPs to pick up activities from the Schedule, it the responsibility of specific Reflection Plugins to pick up stage 3 activities on the bench and apply any required business logic to them before storing them in the Biography. Figure 36 illustrates the relationships of these components. The blue arrows that indicate communication between components are labelled and Table 3 shows the communication they represent.

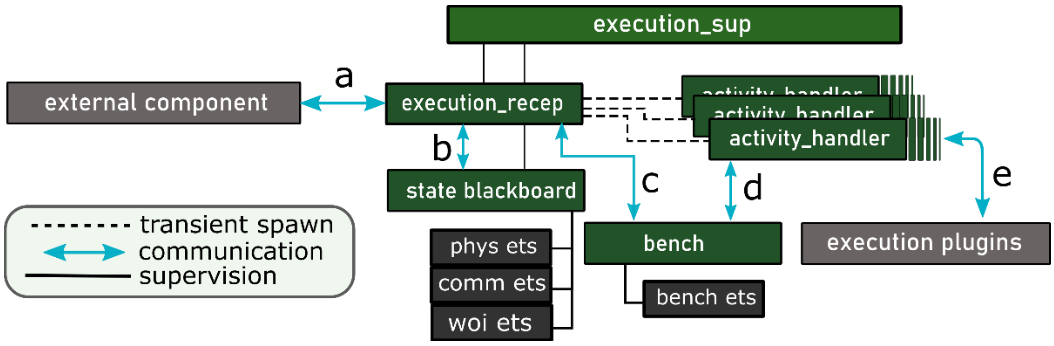



**Figure 36 – Internal structure of the Execution core component**

**Table 3 – Communication details between internal Execution components**

| Label | Description |
|---|---|
| a | External components that want to:<br>• Perform CRUD operations on the state blackboard.<br>• Request a new activity handler for an activity about to be executed.<br>• Query the bench for activities it contains.<br>• Retrieve or remove an activity from the bench. |
| b | Safe CRUD requests |
| c | Safe read/pop requests |
| d | Move completed activities onto the bench |
| e | Activity data read and write requests |

## 7.6 Communications Manager implementation

The Communications Manager consists of the two permanent supervisor and reception processes, along with various other transient processes based on the current communication requirements. The Communications Manager creates transient processes, called *conversations*, that connect and manage communication between the HRH-AS and other holons.

### 7.6.1 Communication Manager as the face of the BASE administration shell

The Communications Manager is responsible for inter-holon communication, as well as answering basic questions on the BASE administration shell and the holon it is part of. The reception component of the Communications Manager is the only component of the BASE core that can communicate with external holons. This allows requests and connections to be controlled and filtered before involving other internal components. The Communication Manager is also responsible for finding and managing BASE BCs of other holons and providing its own BC to others.

### 7.6.2 Communications Manager service conversations

The Communications Manager manages two principle conversation processes, namely: the Service Subscription Conversation (SSCon), when the holon is orchestrating; and the Service Provision Conversation (SPCon), when the holon is being orchestrated. These two processes are two faces of the same communication coin. These processes follow the BASE SPC protocol, as described previously, when two holons enter a BASE SPC.

The Orchestrating Holon will spawn an SSCon and pass to it the BASE BC of the required service-providing holon, along with the initial contract request. The SSCon then sends an RFP to the required holon's Communications Manager. On receiving the RFP, the orchestrated holon is prompted to spawn a SPCon to handle the RFP. A communication link between the two holons are formed, allowing the BASE SPC protocol to be followed and a new contract created or refused.

These conversation processes will remain active until the contract is terminated. Figure 37 illustrates two linked conversations spawned by the Communications Managers of two holons and how EP can use this link to work with other holons during the execution of an activity.

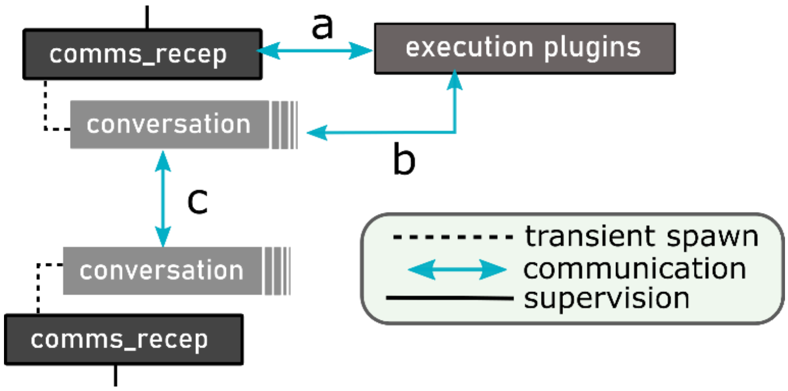



**Figure 37 – Plugins using established service conversations with other holons**

**Table 4 – Service conversations communication arrow labels and descriptions**

| Label | Description |
|---|---|
| a | An EP requests an SPC with a holon for which it has a BASE BC |
| b | The spawned conversation allows the EP to establish a service contract and communicate with the contracted holon. |
| c | The conversation processes of one holon's CM sends Erlang messages to the conversation process of the other holon. |

## 7.7 Handling of 3SAL activities within BASE components

This section describes the handling and implementation of 3SAL activities in a BASE administration shell. The Schedule, Execution, and Biography components take the 3SAL activity through its three life-cycle stages.

Activities in the Schedule, Execution or Biography components can be discovered by querying any combination of shell values. This includes searching for activities by their scheduled, started and completed times within a range or a value; searching for activities of a certain type; or searching for an activity with a specific ID. The result of this query is a list of matching activity shells for activities held by the component. The lightweight nature of 3SAL activity shells allow fast and efficient communication of activities for decisions on holon schedules and histories without requiring potentially large quantities of activity-specific data to be transported. The shell of an activity can be used as its unique identifier when CRUD operations are needed on it.

While an activity is in the Schedule, its first stage data block can be edited. When an activity has been started and is in stage two, only its execution data block can be edited, and when in the Biography as stage three only its third data block can be changed. SPs create and manage stage one activities based on SPCs accepted, or internal activity requirements. EPs request an Activity Handler be created and manages the execution of an activity, as well as notify the Activity Handler when it is completed so it can be upgraded to stage 3 and placed on the Bench. RPs place benched activities in the Biography and manage them for the rest of their existence in the Biography.

# 8 Development of BASE components for human interfacing

This chapter aims to address, in more detail, the two responsibilities of the BASE architecture Execution component – the facilitation of information flow to the human, and information flow from the human. Due to the complexity of the Execution component's responsibilities, further refinement of the component's internal structure was required to support the implementation the BASE architecture HRH-AS. The content of this chapter was published and presented at the International Workshop on Service Oriented, Holonic and Multi-Agent Manufacturing in 2020 (Sparrow *et al.*, 2020).

The chapter starts by identifying two types of interfaces, namely *personal* and *environmental* interfaces. The discussion then focusses on the use of these interfaces to present information to humans and to gather information from humans, through the use of *semiotic* and *observation* services. The refinement of the Execution component's architecture, through the integration of these services, is then described.

## 8.1 Personal and environmental interfaces

I4.0 environments are expected to exploit the advances of smart sensor and interface technologies. These interface technologies can be embedded in equipment, be installed as stand-alone systems or take the form of wearable devices. These interfaces can be used in two forms – as *personal* interfaces or *environmental* interfaces, as shown in Figure 38.

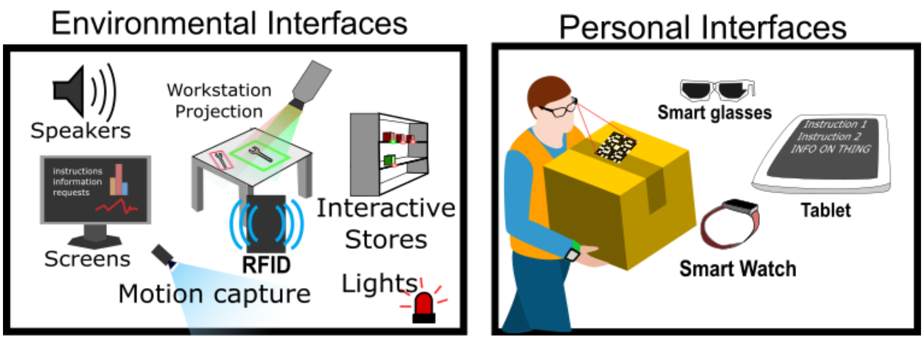



**Figure 38 – Examples of environmental and personal interfaces**

Personal interfaces are maintained by devices that belong to a specific human and are not used by other humans or systems in the surrounding environment. These interfaces can be customized and optimized to fit the specific user and will have a direct connection with the human's associated HRH-AS. Some examples of personal interfaces are those encountered in smart watches, tablets, heart rate monitors, eye tracking devices and cell phones.

Environmental interfaces, on the other hand, do not belong to a specific human. Instead, environmental interfaces are used to gather data from, or present data to, a specified environment. Examples of environmental interfaces are closed-circuit television cameras or digital displays. Environmental interfaces can provide information or services to requesting entities, such as the HRH-AS proposed in this dissertation.

Table 5 highlights the differences between personal and environmental interfaces that should be taken into account by the Execution component when deciding on information delivery.

**Table 5 – Differences between Environmental vs Personal Interfaces**

| Environmental Interfaces | Personal Interfaces |
|---|---|
| **Interactive and adaptive environment**<br>Able to project or augment how the environment is perceived through generalised interfaces. | **High bandwidth**<br>Interfaces offer specialised higher quality rendering of audio, tactile or visual media, as well as availability of other human senses. |
| **Collaborative activities**<br>Information rendered in the environment is available to all collaborating parties at the same time. | **Dedicated interaction**<br>Wearable interfaces can deliver instructions to a specific worker and can be dedicated to a specific function. |
| **Location specific information**<br>Information can be focussed to a specific area or object to reduce information clutter. | **Location independent connection**<br>Information can be delivered regardless of location or visibility of an object. |
| **Low interruption**<br>Workers can choose when to look at information displayed in the environment and can choose when to switch focus. | **High attention**<br>Interfaces that a worker is wearing, or that are augmenting his reality, can demand attention when needed and ensure information is noticed. |
| **Offers implicit communication**<br>Modifications in the environment with low interruption allows information to be conveyed perceived as a "feeling". | **Mainly explicit communication**<br>High attention and high bandwidth can convey information for understanding and acknowledging. |

## 8.2 Information to the human: semiotic services

Advanced interfacing technologies are considered as a key enabler for the Industry 4.0 vision (Posada, Toro, Barandiaran, *et al.*, 2015; Sparrow *et al.*, 2019a). This section describes a means of delivering information to the human with flexibility and robustness through what will be called semiotic services.

### 8.2.1 **Semiosis and multimodal rendering of signs**

Semiosis is the production and communication of meaning through different modalities of signs. Modality refers to the form of media in which the sign is presented. Signs, in the sense of semiotics, is not just pictorials and symbols, but sounds, words, lights, or any stimulation through human senses that represent some meaning to the human (Bains, 2006).

Multimodal interaction enables optimisation and robustness of data delivery, since it allows equivalent information to be presented through different channels (Thiran, Marques & Bourlard, 2010; Baheti & Gill, 2011). For example, multimodal interaction could be achieved at a workstation by providing instructions to a worker via a tablet (through text or sound) and an overhead projector (by highlighting relevant areas of the workspace).

While screens, numerical displays, lights and speakers are widely available technologies that deliver information to the human senses, these technologies are limited by single modality and close vicinity. Smart glasses and head mounted displays are a form of visual Augmented Reality (AR) that display computer generated scenes, and have been demonstrated to facilitate training, stock management, maintenance, and other activities (Peden, Mercer & Tatham, 2016; Quint & Loch, 2015).

Work station projection allows for the display of information directly onto the work place without the need to wear a headset (Bertram *et al.*, 2018; Doshi, Smith, Thomas, *et al.*, 2017). Speech synthesis has seen applications and development ever since the IBM704 sang Daisy Bell for the first time. Modern examples encountered every day are GPS systems and smart phones.

The World Wide Web Consortium standards for multimodal media applications were created to try and consolidate information delivery to humans on different devices and could form the basis on how to expand this ontology to work for other environmental and personal interface types and not just screen based media (Ashimura & Dahl, n.d.).

### 8.2.2 **Holons providing semiotic services**

It is expected that I4.0 environments should be capable of multimodal semiosis through the integration and utilisation of interfacing technologies. Should these environments be represented as holonic systems, these interfaces would be integrated as Interface Resource Holons providing semiotic services. Each

Interface Holon will be specialised in its particular modality to optimise and personalise the delivery of a requested piece of information.

Apart from the Resource Holon Responsibilities presented in Section 2.3.1, holons providing semiotic services also perform responsibilities pertaining to:

- Owning, managing, and controlling its physical rendering component (e.g. screen, speaker, projector etc.); and
- Optimising the information delivery using knowledge of its modalities and data given to it on the targeted human.

Various Interface Holons providing different modalities to deliver information will provide robustness and redundancy. The interfaces will have the ability to optimise the information delivery using the State Variables and Attributes. This will allow a manufacturing environment with drastic improvements in human situational awareness, safety, and overall connectivity with the factory's digital environment.

## 8.3 Information from the human: observation services

Since humans have no means of digitising and communicating their own state in real-time, they require dedicated systems to take up this responsibility. The Execution component of the BASE architecture is responsible for providing digitised version of the human's state. The Execution component will make use of observation services to gather data.

### 8.3.1 Measurement and fusion of state variables

Obtaining accurate measurements of dynamic systems with complex behaviour, such as human workers in manufacturing environments, is a challenge. As such, the measurement values should be considered along with associated confidence values. Confidence, as a representation of uncertainty, can be expressed in different ways – as standard deviation, intervals, or accuracy and precision pairs.

Confidence deteriorates over time; therefore, this research proposes that a timestamp be added to data obtained from sensors or calculated by algorithms. Data will thus be presented as a Value-Confidence-Timestamp (VCT) triple.

Figure 39 illustrates a motion capture observation service reporting the position of a worker as well as the angles of his upper arm and elbow, along with the associated confidence scores of the measurements. The confidence is illustrated as distribution of colour where a high intensity of red indicates high probability.

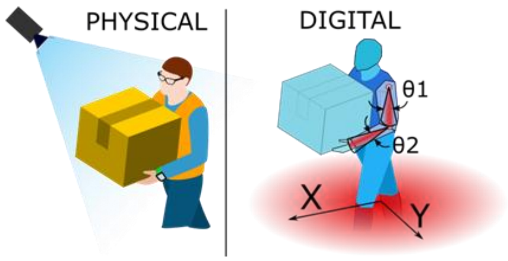



**Figure 39 – A VCT state variable taken for position and two limb angles by a motion capture camera showing high confidence as high intensity of red.**

Table 6 shows what the VCT measurements may look like for the illustration in Figure 39. Observation services should deliver observed data in VCT format to ensure that the Execution component can fuse the measurements from multiple sources to provide a single source of truth on the human's state.

**Table 6 – Examples of how VCT measurements may be captured on a human**

| Variable | Value | Confidence | Time |
|---|---|---|---|
| X position | 23.3 m | 0.85 | 10:23:05 |
| Y position | 23.3 m | 0.91 | 10:23:05 |
| Shoulder angle | 14 deg | 0.73 | 10:23:05 |
| Elbow angle | 95 deg | 0.65 | 10:23:05 |

### 8.3.2 **Holons to provide observation services**

Within a holonic system, it is assumed that physical sensors will be part of a holon that advertises and performs observation services. Other holons in the system, such as the HRH, can then obtain and use the information from these observation services. Apart from the Resource Holon Responsibilities presented in Section 2.3.1, holons providing observation services need to fulfil the following specific responsibilities:

- Own and manage its physical sensor components; and
- Refine and interpret the sensor data to provide its observations in VCT format.

## 8.4 The Next-Execute-Update protocol

In *Design for the unexpected* by Valckenaers & Van Brussel (2016), the Next-Execute-Update (NEU) protocol is presented as a means to facilitate cooperation between an entity with process knowledge and an entity with the ability to execute process actions. The NEU protocol decouples the decision-making on process steps from the executor of the process steps.

The NEU protocol starts by giving an instruction to the executor (the Next step), the executor carries out the instruction (Execute), and the decision-making entity then determines the next appropriate execution step to be instructed to the executor (Update). The NEU protocol thus facilitates effective cooperation between the human (executor) and the EPs (decision makers). The EPs will request instructions to be communicated to the human worker who will proceed to execute them. The EPs then monitor the state of the human and their WOI to decide on the next execution steps and instructions to communicate.

## 8.5 Execution components for Interfacing

Sections 8.1 through 8.4 introduced mechanisms, concepts and protocols that are considered critical to the development of the Execution component of the BASE architecture. During the first implementation of the architecture, the components described in this section were incorporated into the Execution component of the BASE core, since the architecture was initially only intended for the human holons involved. Subsequently, the BASE architecture was generalised to act as the administration shell of any resource involved, and the human specific components described here was generalised to EPs.

### 8.5.1 **Architecture overview**

The Execution component is responsible for capturing information from and presenting information to the human worker. The Execution component performs these functions through observer and semiotic services, by making use of personal and environmental interfaces.

The use of observation and semiotic services is managed by two architectural elements within the Execution component, namely the *Observer* and *Informer*. The architecture provides a mechanism to gather and display the information within the Execution component, which is implemented as the State Blackboard. The Execution component’s internal architecture is illustrated in Figure 40.

In later iterations of the Execution component, the Observer and Informer were changed to EP components in order to generalise the architecture for use on other resources. The State Blackboard is not human-specific and forms an integral part digitising the state of any resource that a BASE administration shell is applied to. Therefore, the State Blackboard would remain as part of the Execution core.

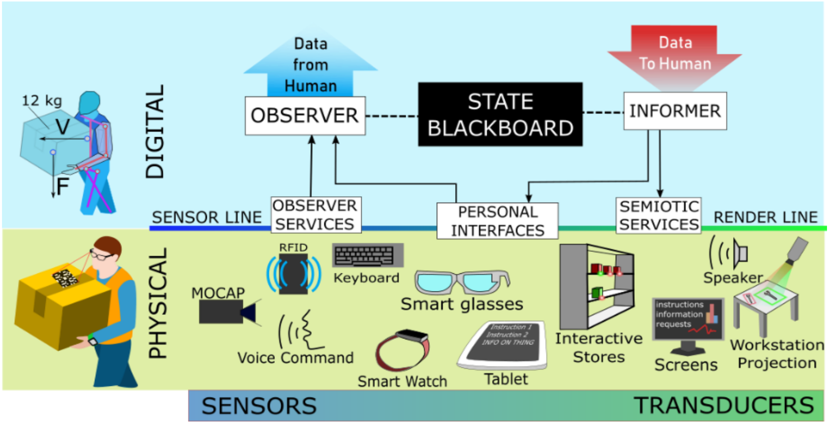



**Figure 40 – Execution interfacing between the physical and digital world**

### 8.5.2 The State Blackboard

The State Blackboard (SBB) serves as a synchronous, single source of truth on the HRH's current state. The human's current physical, mental, and biological state is updated by a modular component called the Observer, discussed in Section 8.5.3. The SBB also keeps track of the WOI, as specified by the ARTI architecture. This ensures any critical data for execution, as well as safety monitoring, will be available to the components of the HRH-AS and can be communicated to external holons.

### 8.5.3 The Observer

The Observer has the responsibility of providing a digitised version of the physical human state. This is accomplished through the gathering of information from observation services, and the derivation information using the SBB and the HRH Attributes. The Observer handles any information received from the human through interfaces and interprets it according to the operating context, as indicated by variables on the SBB and Attributes. Furthermore, since all observational services will deliver the state data in VCT format, the Observer can consolidate data from different sources into a single source of truth.

### 8.5.4 The Informer

The Informer delivers information to the human. It uses the Attributes to personalise the information delivery and uses the data on the SBB to optimise how it's delivered through different semiotic services. It is important to note that the Informer does not decide what the human should do – it is only concerned with the intelligent delivery of requested information.

# 9 Case Studies for evaluating the BASE architecture

To demonstrate and evaluate the BASE architecture against its responsibilities to facilitate the effective integration of human workers, two case study implementations and evaluations were performed. This chapter details the objectives of each case study and the selection of an industry partner with which the case studies were conducted. The evaluation, based on these case studies, is presented in Chapter 10.

## 9.1 Overview of case studies

The case studies aim to demonstrate that the BASE architecture fulfils the three responsibilities of an HRH-AS described in Chapter 4, namely:

- Delegation on behalf of the human to other digital holons.
- Interfacing with the human using environmental and personal interface services.
- Managing digital data and processing on behalf of the human.

With the fulfilment of these responsibilities, it will be demonstrated how the BASE administration shell elevates the human worker's status to the level of resource holon, effectively forming a HCPS in the form of a Human Resource Holon.

The first case study focusses on the ability of the BASE architecture to facilitate the interfacing of humans and their digital environments. The first case study also serves as a technology demonstrator – enabling a subsequent case study in an actual manufacturing environment to test the ability of a BASE HRH-AS to integrate factory workers with other I4.0 enabled assets, through the management of the involved AL.

### 9.1.1 **Selecting a case study partner and scenarios**

The selection of an appropriate case study was important for demonstrating the functionality of the BASE architecture. After considering the processes of local industries, a layup activity of a local composites manufacturer was selected as the case study for the following reasons:

- The processes of the composites industry are labour intensive, with automation often not possible or feasible.
- The local industry partner is a supplier to the aerospace industry, which requires meticulous traceability and quality assurance.

- As a typical South African company, the chosen industry partner spends significant resources on training an under-skilled workforce.
- In the composites industry, early adopters of new technology often enjoy a competitive advantage – the industry partner was thus interested in the potential benefits that the implementation of the BASE architecture has to offer.

The Head of Innovations and Technology at the industry partner served as liaison and aided in choosing an appropriate layup activity to replicate and helped to identify three benefits that the industry partner would like to realise with an implementation of the technology. These benefits would guide the plugin development for the case study and provide the value addition to the data handled by the BASE HRH-AS. The three benefits are as follows:

- **Improved traceability** of process deviations and an improved communication chain between the shop floor and the engineering division. Particularly through the ability of workers to record process problems during an activity with an image they took of the problem along with pose and position recordings of the workers.
- **Automated Standard Work** measurement with time and motion analysis. Current time and motion studies are difficult to conduct and require time to plan and execute. Furthermore, the measurement process is invasive and can cause skewed results.
- **Worker activity process facilitation** through use of digitised work instructions and reactive process advice through either a tablet or an overhead workstation projector.

These benefits were previously limited by the AL required for their realisation and it was expected that the implementation of the BASE architecture would enable these goals to be reached.

### 9.1.2 Case study objectives

Case study 1 aimed to evaluate a BASE implementation's ability to support the interfacing of humans with their digital environments. For this case study, a layup activity was recreated in the Automation Laboratory in the Department of Mechanical and Mechatronic Engineering. The laboratory setup allowed for different interfacing service holons to be created and tested with a BASE administration shell.

The industry partner's liaison was consulted on the structure and direction of the case study so that it closely resembled a potential on-site implementation in the future. The other reason for conducting the first case study in the laboratory is due to the possible disruption that various hardware items may have had on the productivity of workers on the shop floor when testing different new interfacing capabilities.

The case study's objectives in evaluating the BASE architecture were as follows:

- Establish that the BASE architecture implementation is capable of orchestrating interfaces available to the worker.
- Demonstrate that the BASE architecture implementation can manage the data involved with an activity's lifecycle by the four types of plugin components.
- Show that the system requires no disruptive training or effort from workers that could disrupt an on-site implementation.

Case study 1 aimed to validate that an implementation in an actual manufacturing environment is feasible, and case study 2 focused on evaluating if a BASE architecture implementation would have a positive effect on the AL involved with business processes that involve human workers, and thus facilitates their integration with a modern manufacturing environment.

Case study 2 involved observing existing AL Processes (ALPs) in the operations of the industry partner, which are used to manage human workers, their activities, and the resources involved. The aim of this observation was to measure and evaluate the effectiveness of the ALPs. The same workers and resources were then equipped with BASE administration shells and the AL effectiveness was measured and compared to the current, normal operations of the industry partner. The objectives of the second case study were to:

- Evaluate the difference in ALP effectiveness for a chosen set of business processes.
- Explore the experiences that workers have with the BASE HRH-AS.
- Compare the difference between setting up a new ALP for capturing data from an activity with and without a BASE architecture implementation in place.

The objectives of this case study for the industry partner, which would guide the plugin development for this implementation, were similar to the objectives of case study 1. However, the objectives were focussed on the existing processes involved in layup activities and address specific issues related to the shop floor.

## 9.2 Case study 1 – Interfacing with humans

### 9.2.1 Case study description

The layup activity (LUA) begins with a worker receiving a worksheet with the layups that need to be completed for the day. When a worker starts a layup activity, they fetch the required ply pack and mould for the layup. A ply book is supplied for each layup activity containing ply order, placement, and mould information. A worker will then receive the mould and start the layup process by unpacking the ply pack. Each ply is sequentially placed in the mould with the use of plastic spatulas and heat gun, to make pieces more malleable.

Most layups require the addition of core-filler, a hardening polymer substance used to fill up parts of the honeycomb. The core-filler is applied using a template showing the correct areas to be filled. The same component can have different core-filler templates, and the worker needs to make sure the serial number of the template matches that specified on the worksheet.

After the final plies are laid up, the part is placed in a vacuum bag and taken to the vacuuming station – marking the end of the layup activity. A simplified version of this process was created to replicate key components of the layup process, and a flow diagram illustrates this in Figure 41.

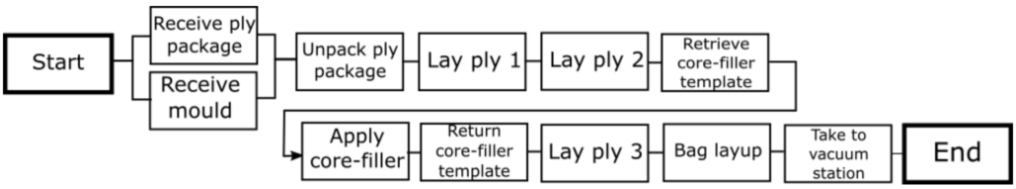



**Figure 41 – Layup activity action plan**

### 9.2.2 Plugin development and customisation

This section discusses the development of the plugins to the BASE architecture to adapt it to the purposes of the case study. Along with the business decisions reflected by the plugins, the architecture will be customized with the required State Variables and Attributes needed for the plugins to make appropriate decisions.

Table 7 lists the State Variables that were required for making scheduling and execution decisions. The State Variables were separated into three categories: Environmental State Variables (ESVs), representing the WOI of the HRH; Physical State Variables (PSVs), representing the digital reflection of the HRH's physical state; and Communicative State Variables (CSVs), for information the human wishes to communicate to plugins or other digital systems.

**Table 7 – Required State Variables**

| State Variable | Type | Description |
|---|---|---|
| Ply State (PLY_STATE) | ESV | Determines which of the ply locations on the table are occupied. EPs will use this ESV to make process state decisions. |
| Core filler template position (TEMPLATE_STATE) | ESV | Like the plies, the core filler template state will be used by the EPs for decisions as well as recorded for the Standard Work analysis. |
| Layup table state (LT_STATE) | ESV | In order to see if it is viable to start a layup activity, the status of the layup table needs to be known. |
| Pose data (POSE) | PSV | To determine which step of the layup activity the worker is busy with as well as recording the pose for time and motion analysis. |
| Location data (LOCATION) | PSV | The execution tools need to determine where the worker currently is on the shop floor for communication and process decisions. |
| Activity proposal (RFP_RESPONSE) | CSV | The human's response to an activity request. |
| Ready flag (READ) | CSV | The response from the human to a query by the EPs on if the activity can begin. |
| Error flag (PROCESS_ERR) | CSV | Variable to communicate an error in the process for the appropriate plugins to respond to. |
| Error description (ERR_DESCR) | CSV | The description of an error from the human's perspective to be received and managed by Eps. |

Table 8 shows a list of Attributes required for use either by the decision-making components or the administrative functionality of the core components.

**Table 8 – Attributes required for the case study implementation**

| Attribute | **Description** |
|---|---|
| NAME | The name of the human |
| BASE_PATH | The path to the stored BASE profile |
| LUA_DUR | The average duration of a layup activity. |
| LUA_STDWRK | The average time of the component actions for a layup |
| LUA_COUNT | The amount of layup activities performed by this HRH |

#### 9.2.2.1 RFP Scheduling Plugin

One SP was developed to handle the RFP the worker will receive when they are required to complete a specific layup activity. This RFP handling SP creates an internal activity on the Schedule that will request the human to decide on the time they propose to complete the activity. The response of the human is used by this plugin to send back a proposal to the requesting holon.

#### 9.2.2.2 LUA Execution Plugin

An action plan was first created to pair instructions, process steps and observations for the LUA, shown in Appendix C. The LUA EP will start a layup activity that was scheduled and perform the following two functions:

- Guide the human through the activity process steps and orchestrate the other resources that are required for the activity to be completed.
- Capture key activity data into the execution data block of the activity.

The plugin was implemented using a Finite State Machine (FSM) that reflects the layup process diagram, as shown in Section 9.2. The FSM reflects how the process is visualised by process planners and workers. To aid in its development, the DRAKON Erlang state machine editor by Mitkin (2018) was used. DRAKON is a visual flow diagram tool that outputs the constructed state diagram as an Erlang FSM.

Figure 42 shows one of the ten activity states illustrated in Figure 41 - the Unpack Plies state. The Unpack Plies state diagram exhibits three branches: for successful action, for process errors and for unknown events. If the Error State is entered, the plugin will request signs to be rendered through available interfaces to instruct the human to provide a picture and a description to be captured in the execution data block for that activity.

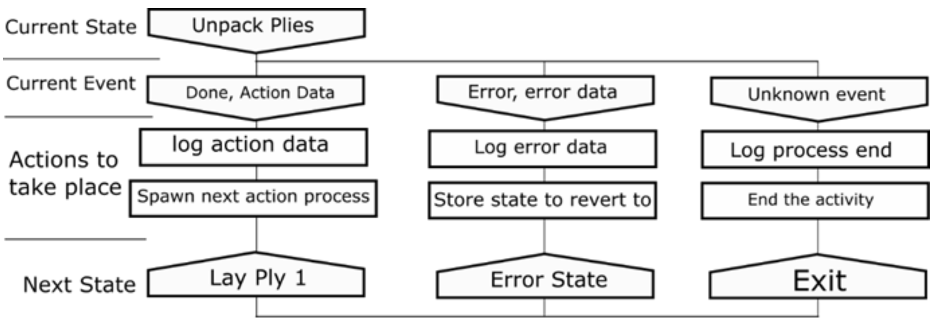



**Figure 42 – A portion of the DRAKON Execution Plugin state machine showing the unpack-plies state flow**

#### 9.2.2.3 RFP Activity Execution Plugin

A similar state machine was created for an EP to capture inputs from the worker about activity requests being handled by the RFP SP. This Plugin asks the human to propose a time to complete a requested activity - the reply is written as a CSV with a unique tag. When the tag is detected by this plugin, the activity is considered complete. The reader may recall that this reply is also used by the RFP SP to complete the proposal sent to the orchestrating layup activity holon that requested it. An example of this activity being executed is given in Section 9.2.4 as part of the evaluation.

#### 9.2.2.4 Pose Logging Activity Execution Plugin

The Pose Logging Activity EP subscribes to the Pose Observer Holon, and records this pose information during the execution of a layup activity. The pose data is stored in an XML file, with the location of the file recorded in the execution data block of the activity. This plugin was created to fulfil the industry partner's goals of automating the data capture for Standard Work and motion analysis.

#### 9.2.2.5 Activity Initiator Execution Plugin

It was chosen to allocate the responsibility of checking the Execution Gate of the Schedule to one EP that could then instruct other plugins to react accordingly. This plugin was implemented as a single Erlang process that would poll the Schedule for gated activities once every second and start the appropriate EPs when an activity is due. This plugin was called the Activity Initiator.

#### 9.2.2.6 Reflection Plugins

In the case of this implementation there were no post-execution data sources and the RP only created the Biography entry. The RP was implemented as a single Erlang process for each type of activity described previously. An Erlang process, called the Bench Check, polls Execution for completed activities once every second and then spawns the appropriate reflection process for the completed activities it found. The reflection process for an activity creates the Biography entry for the completed activity.

#### 9.2.2.7 Standard Work Analysis Plugin

The plugin to analyse Standard Work was a single Erlang process set to retrieve all completed layup activities in the Biography for the day. It then calculates the average time for each action during the activity, and stores this as a new LUA_STDW Attribute.

#### 9.2.2.8 Motion Study Analysis Plugin

An AP that creates a visualisation of the worker's movement and pose throughout completed layup activities was created. All activities could be used to generate a heat map of motion to gain insight into the average movements involved in a layup activity, or an individual activity could be referenced from the Biography and its pose log visualised.

### 9.2.3 Observation and semiotic service holons

The case study aimed to demonstrate the ability of the BASE architecture to interface humans with their digital environment through orchestration of interface service holons. The interface holons were only developed to the extent where communication with the HRH could be demonstrated and basic services could be executed. The complete development of sophisticated holons was thus considered outside of the scope for this case study implementation.

The interface holons were implemented with a generic administration shell, in accordance with RAMI4.0, and customized according to the service that is performed. The administration shell handles communication with the HRH-AS, while managing the physical sensing or rendering device. It was not required to test the full holonic functionality of these interfaces, since their modularity and communication with the BASE administration shell was all that was needed. Therefore, scheduling and virtual execution is not implemented in these holons, nor was much emphasis placed on their intelligence.

Interface holons were implemented using NodeJS and JavaScript based HTML5 web applications, which were chosen for the following reasons:

- HTML5 provides modularity and hardware redundancy. Any computer, phone or device capable of running HTML5 can become the resource with access to its sensors and cameras.
- JavaScript provides a quick means of producing simple programs with multimedia components.
- Large supporting community with existing tools.
- NodeJS provides a simple webserver platform with nearly effortless expansion of functionality using community developed modules.

An HTML5 capable device logs into the NodeJS server acting as the administration shell. The device then becomes the physical resource controlled by the administration shell of a particular service – as illustrated in Figure 43, where the HRH-AS connects to the holon through its NodeJS server. Table 9 lists all the developed interface holons, with the implementation details presented in Appendix B.

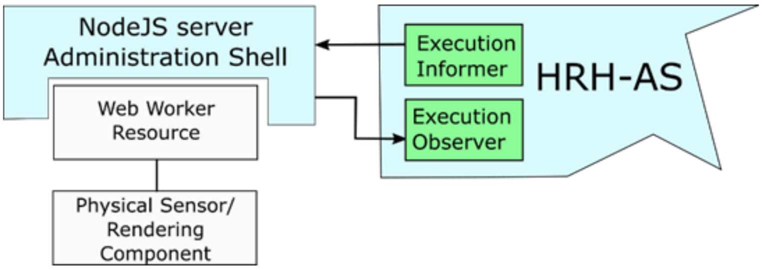



**Figure 43 – General interface holon structure**

**Table 9 – Developed interface services for BASE case study implementation**

| **Interface holon** | **Service Description** | **Interface Description** |
|---|---|---|
| Pose Observer | Provides pose data on a worker. | PoseNet (Neural net pose identifier) [4] runs on the browser of a computer or HTML 5 capable device with a webcam sensor component. |
| Layup Table State Observer | Provides the state of the layup table, the plies and the mould. | P5JS library with a HTML5 capable device and webcam sensor component |
| Tablet Interface | Provides text, schedule, and activity request rendering as well as text and image input from the user. | LibGDX game engine with android, touch screen and camera API |
| Ply Projector | A augmented reality visual semiotic service that can overlay signs on the ply table | HTML5 capable device with a projector |

### 9.2.4 Lab setup

The replicated layup station in the Automation Lab made use of two tables. One was used as the layup table, and the other as the template store. Three areas of the lab were designated and identified as "ply table", "template table", and "vacuum station" (the vacuum station is not visible in Figure 44). A 3D model of the lab was created and is used to show the layout of the layup station in Figure 44.

[4] PoseNet is an open-source tensorflow based model for detecting human poses in real time from a webcam feed.

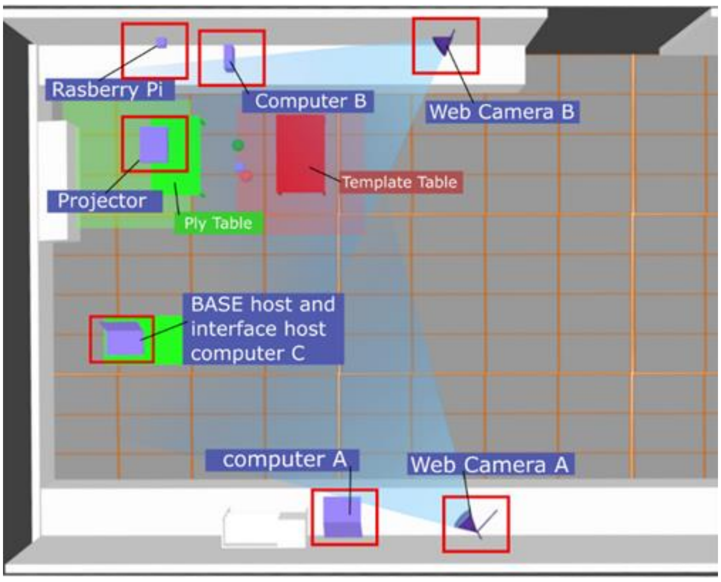



**Figure 44 – Top down view of lab setup and the devices involved**

Two web cameras, connected to computers, were placed on either end of the lab to provide a full view of the three specified locations in the work area. The camera axes were placed orthogonally to minimise error on the position triangulation (Wrobel & Wolfgang, 2001). The cameras act as the physical sensing devices controlled by the Pose Observer holon. Table 10 identifies all computing and interfacing devices in the area.

**Table 10 – lab setup device list and description**

| Device | Description |
|---|---|
| Computer A | Acer Aspire 5750 running Chrome browser |
| Computer B | Dell OptiPlex 980 running Firefox browser |
| Computer C | Dell Inspiron 15 7000 running Chrome Browser |
| Web Camera A and B | Generic 720p USB web-camera module |
| Raspberry Pi 3 | Linux distro with chromium browser |
| Projector | Epson XBS 11 |
| Tablet | BEON IQ NT7W running Android 5.1 |
| Smart Phone | Xaomi Redmi Note 5 running Android 7.1 |

### 9.2.5 **BASE HRH-AS operation**

This section describes the operation of the BASE architecture HRH-AS implemented in the case study. First an overview of the operation is presented with a diagram illustrating the flow of an example LUA, then the LUA and other activities that follow the example flow is discussed.

9.2.5.1 Operation overview

The operation and interaction of the BASE architecture components, as shown in Figure 45, are described for four different activity types that were considered in this case study, namely: the layup activity, the pose log activity, the RFP activity and the LUA analysis activity.

To fully visualize the emergent process that interacting BASE components create, Figure 45 shows a simplified overview of the layup activity data life-cycle and attribute updating, and the completion of the cycle with a newly scheduled layup activity. When viewing Figure 45, the reader is encouraged to start with the SPs and proceed in a clockwise manner. Any other activity that uses persistent data from its HRH executor is expected to benefit from the same process.

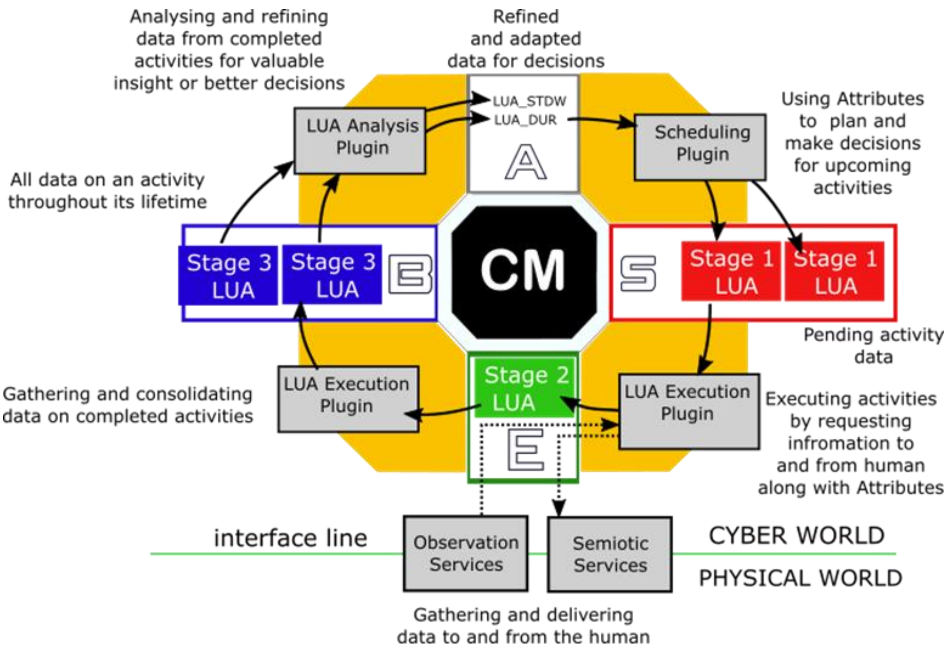



**Figure 45 – Overview of the interaction and flow of information between the BASE architecture components.**

9.2.5.2 Orchestrating interfaces available to the worker

This section describes how the BASE HRH-AS successfully managed the orchestration and data handling required to interface with the human worker through interface service holons. Images showing the HRH-AS reactions through semiotic services, based on changes to the HRH's state detected by observation services, provide evidence of the successful information gathering and delivering with a human worker.

Reacting to environment and human state changes

The LUA EP that utilises a state machine to guide the worker through the execution of a layup utilises the pose, position, and table states presented on the SBB by subscription to the Pose Observation holon, Table State Observation holon, and the worker's tablet.

Figure 46 presents three images from the execution of a layup activity that show the workstation projection and worker tablet changing, based on the activity and worker states. The instructions for the next process steps are displayed on the work station, as well as on the tablet linked to the worker's BASE administration shell.


**Figure 46 – LUA in execution with the first three steps and their time stamps from the execution data block shown.**

Reacting to human communication

On detection of an error during the activity, the worker can inform his administration shell by pressing the Error button on the tablet interface. Table 11 shows the data from a process error event captured during the completion of a layup activity. The worker identified the pack was missing ply number 3 and pressed the error button, which caused his tablet interface to request an Error CSV be written to the SBB.

**Table 11 – A process anomaly report extracted from a stage 3 activity.**

| Item Description | Item content |
|---|---|
| **Issue type** | Material Issue |
| **Image of issue** | event_M_user_17611113_19_12_07_11_01.jpg |
| **Worker description** | Ply 3 Missing from pack |
| **Time of occurrence** | 1562533954817 |
| **Time of resolution** | 1562533970826 |
| **Activity Shell** | {"LUA",1562532523828,1562533841335,1562533989303} |

The LUA EPs reacted to the appearance of the error variable by initiating the error handling sequence, asking the worker what the error was related to – as shown in Figure 47 with a screenshot of this dialog. More images showing how the worker is guided through the process steps using the various interfaces available can be seen in Appendix E.


**Figure 47 – Anomaly handling dialog on the tablet interface**

The worker then pressed the material error button, and the tablet opened the camera to allow them to take a picture of the situation. The image was stored on the tablet and the image name, along with a possible comment from the worker, was written to the SBB as CSVs, which the EPs used to compile the report that is appended to the execution data block of the LUA.

The RFP received by the SPs create a Layup Request Activity (LRA) that is intended to query the human on a proposed time to do the requested layup. The SPs have access to the Attributes and State Blackboard, and can thus wait for the successful execution of this activity and the subsequent CSV to appear on the blackboard to complete its proposal.

The sequence of images in Figure 48 describes the activity request appearing on the tablet, being orchestrated by the EPs, proceeded by the user dragging and dropping the request onto a timeline where the time is then recorded to the SBB by the execution plugins. The images also show the duration of the activity as retrieved from the HRH's Attributes as the LUA_DUR value.

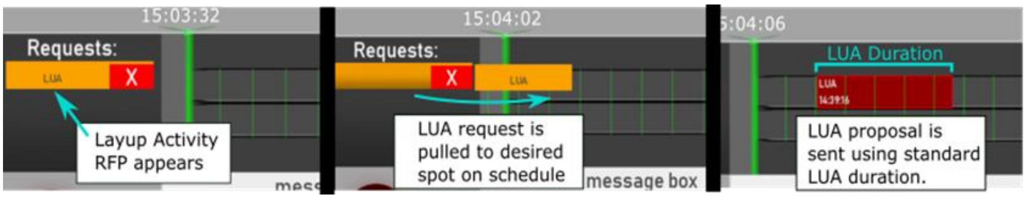



**Figure 48 – Screenshot demonstrating the activity request handled by a worker**

9.2.5.3 Manage the data involved with an activity's lifecycle

Each scheduled LUA triggered the scheduling of a Pose Logging Activity (PLA) during the same time period. The PLA EP requests pose data from the SBB and logs it to a file. Once a LUA is completed, the PLA is also completed, and the pose log file location is recorded in the PLA's execution data block.

This case study demonstrated the ability to automatically capture critical process information from a layup activity. This information included error reports, time and motion data per activity, as well as a log of all activities performed.

A tool that visualises the captured time and motion data of the Pose Logging Activities was created – an example of typical output shown in Figure 49. Green dots are the left hand, blue dots are the head, and red dots are the right-hand recorded positions of the worker during execution of a layup activity. The black lines show which sets of dots are neighbours in time, as to visualise the movement of the worker through the environment.

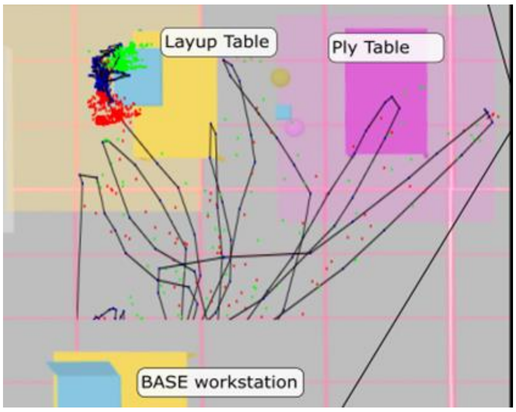



**Figure 49 – Activity Pose Visualiser Analysis Plugin sample output**

Figure 50 shows a time sequence visualisation of the duration each action took during the layup activity. Figure 51 shows the trend of Standard Work for the activity over 6 activity attempts. It was deemed plausible to implement BASE administration shells on the industry partner's shop floor. This will allow multiple workers and layup activities to form part of the evaluation.

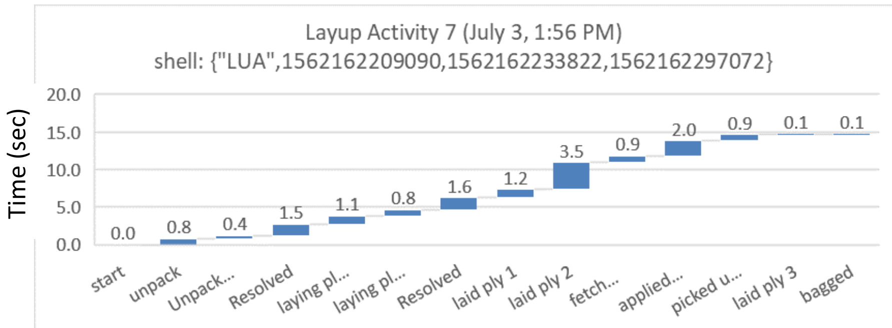



**Figure 50 – Action time visualisation from captured execution data**

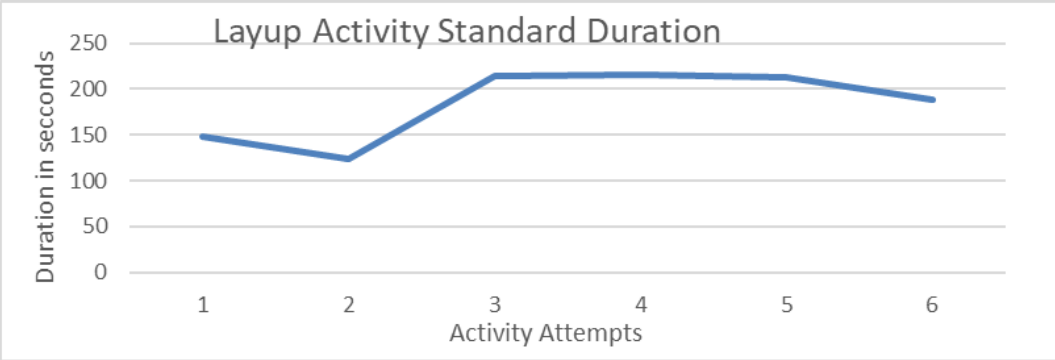



**Figure 51 – Standard layup activity duration attribute over the course of six layup activity attempts**

## 9.3 Case study 2 – Managing Administrative Logistics

The second case study was aimed at evaluating the effects a BASE administration shell would have on ALPs involved with human activities. This case study, along with the first one, will evaluate how well a BASE HRH-AS will facilitate human integration in an I4.0 manufacturing environment.

The case study observed various business processes with different decision-making requirements and their accompanying ALPs. First, these processes were observed as they took place during normal operation in the production lines. Thereafter, BASE architecture plugins were developed to automate or facilitate aspects of these processes. The ALP effectiveness is then compared between the existing and BASE enabled scenarios.

Two scenarios where chosen where human decision making plays a critical role in the successful day to day operations of the plant. The first scenario considered ALPs over short time spans, and the second considered ALPs over longer time spans due to the differences in how time affects different components of an ALP. The first scenario involves the monitoring and managing of moulding activities. The second involves quality improvement and process visibility.

Each scenario and the observed business processes involved are first described how they are normally accomplished and then how they take place when resources are BASE enabled.

### 9.3.1 Scenario 1 – Monitoring Moulding Activities

Throughout the day, the line lead of a production line needs to know the status of activities that were scheduled and manage problems that may occur during their execution. To accomplish the activity monitoring responsibilities, the Line lead needs to receive information on activity errors and activity status. The result of these decision needs to be communicated to the workers and data storage mechanisms.

When the Line lead required activity status information, the following questions are examples of what would commonly be asked:

- What is the progress of a specific activity?
- What activities are currently in progress?
- Can another activity be given to a worker for the day?

The Line lead would obtain this information through one of two methods. The first method is to walk to the moulding shop and see the progress of different activities or speak to the moulders in the shop – referred to as the *walk-to-see* method. The other method is to use the moulding WhatsApp group and wait for a response from the correct worker. The Line lead will often use the latter method if he is too busy to afford the walk to the moulding shop.

Data can also be pushed to the Line lead when a worker notices something wrong during the execution of an activity, such as a missing ply-pack or a tooling error. The worker has the same observed AL mechanisms available. He can either use the moulding WhatsApp group or walk to find the Line lead.

In both cases, the type of information that needs to be communicated was simple data values. Walking to see the progress on a particular part or reporting that there is a material or tooling error on a specific part requires only a few strings of data. The Line lead needs to walk to the moulding shop only if he deems a reported error critical and worth his time, otherwise walking to see is very often overkill for the information he wants to obtain. The Line lead's smartwatch reported him walking over 12 000 steps on one of the days.

Figure 52 illustrates the data flow into the Line lead where the decisions are made, and the data flow from the Line lead to either the moulders or storage. The AL Processes are labelled as AL_X.X and represent the process for the Line lead to obtain or deposit a specific piece of information.

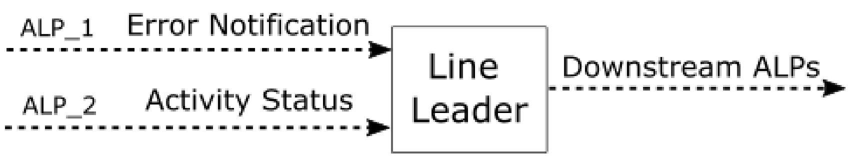



**Figure 52 – ALPs involved with line leader decision making**

### 9.3.2 **Scenario 2 – Quality improvement and process visibility**

The observed composites products take several days to complete and goes through four process steps. The first step is material preparation, which includes kit cutting and consumables. The second is moulding, the third curing, and finally quality assurance and shipping. Due to the low volume and time-consuming nature of the products involved, it is often difficult to spot process trends or make definitive links to events of one day, and discoveries days or even weeks thereafter. Two aspects of longer-term AL will be looked at – process visibility and quality improvement based on the accuracy and visibility of data related to these cases.

#### 9.3.2.1 Worker feedback

Speaking to workers on how they receive feedback on work they have completed, two ALPs were reported to obtain this information. Workers can walk to quality assurance and packaging and look for their parts to get feedback from workers there. Due to the time this takes, only one worker reported to regularly doing this. If a quality issue trend is noticed by either the Line lead or quality assurance, a worker could be informed that a problem with their work exists. This requires active engagement and time from the Line lead that may only have had to give a tip on how to improve. In more serious cases that may require a reproduction of the affected moulding activity, the relevant information on the failure and reproduction requirements will need to be available. Both methods require walking to the source of information and the worker's memory of when and how the activity was completed or if there were any problems with material, tooling, or process that he can remember.

Obtaining process visibility is done by analysis of recorded activity data and event reports. This type of data is used to produce information such as Standard Work. Managing assets, such as tooling maintenance and life-cycles, was also observed to be an important part of the composite process management that requires long term AL. Material out-life is an important part of the Line lead's decision-making processes and tracking and tracing of this information was observed to take effort from both the workers and the Line lead.

#### 9.3.2.2 Obtaining Standard Work data on activities

Companies with product cycle times in the order of minutes may be able to obtain Standard Work data, along with standard deviation, within one or two days of conducting a time study. For the parts observed, such a detailed analysis on their activities may take months out of their industrial engineering team's time and thus a single time study per part is usually conducted for the year and used for all planning and scheduling.

Keeping track of individual activity durations would require reliable and dedicated recording from workers of their activities, and time to digitize this data for analysis. It can be speculated what the cost and accuracy of this may be, but for the intents of the case study this operation can be considered currently impossible due to the cost of the AL involved.

Annually recorded Standard Work times are kept in a spreadsheet. If the Line lead needs to optimize a production plan he needs to find this spreadsheet, look up the values he needs, and manually enter them into a Yamazumi-like tool. The Line lead indicated that the actual decision making is a fraction of the time of the ALPs to prepare the data to work with. The reported average time spent finding data on spreadsheets are between 5 to 10 minutes.

#### 9.3.2.3 Tooling maintenance

A mould used for a layup needs to be cleaned after three uses. This information is stored in the form of a spreadsheet in the moulding workshop. It was observed that this ALP took on average 30 seconds and does not include the time spent fetching a new spreadsheet if the previous one was full. This ALP requires that workers (who may be under time pressure) remember to enter this information on the spreadsheet, as well as notice when a mould is due for cleaning.

The Line lead needs to walk to the workshop and read the spreadsheet to obtain this mould information. The Line lead can also use the moulding WhatsApp group and wait for a reply from the appropriate worker. It was also reported that keeping track of total lifetime uses of a mould becomes a difficult task, since the data on the mould-use spreadsheet needs to be digitised regularly, while assuming all uses were accurately recorded. Decisions on when it needs to be replaced need to be balanced with production orders and the time of acquiring a new mould.

#### 9.3.2.4 Production history visibility and analysis

Data captured on completed layup activities at lies in a combination of the routing labels recorded by the Enterprise Resource Planning (ERP) system, SAP, and the memories of the workers involved. The ERP is only able to track the progress of an activity at digitizing checkpoints or when information on an activity is entered at the end of a day. There is no other method of storing activity events and which resources were involved other than the memories of workers. When this data is required to resolve a quality or lead time issue, all staff spoken to reported it to be a painful process of reconstructing events from workers, spreadsheets and pulling data from the ERP system. Manually recording this data is not beneficial due to the man hours involved in the AL and these issues are therefore dealt with only when they appear.

#### 9.3.2.5 Implementation overview

In this case study, a BASE administration shell was given to all assets involved in the layup activities and the plugins were created specifically for the business processes involved.

The BASE architecture is meant to aid in AL Processes and cannot force a person to think or consume data it makes available. It is thus considered that when data makes it to a BASE HRH-AS and it has the plugins to present the data to a person, the ALP has been completed – the decision to consume the data is left to the person. This section will describe the AL and decision-making processes involved with scenario 1.

For a LUA to take place, a moulding request is made to the LUA type holon. To fulfil the moulding request, the LUA type holon employs the services of a mould holon, ply pack holon, and moulding worker holon to form a LUA activity instance. The LUA type Holon finds the Business Cards of holons that can supply these services during the scheduling phase, and the Line lead aids it in selecting which resources should be responsible for which activity.

From the request for a specific moulding until its completion, the LUA Type holon keeps record of the holons it employed to fulfil the request. The scheduling of moulding activities formed part of the business processes with which the BASE administration shell was involved, but only the execution of the LUA instances where the activity monitoring is involved is discussed here. The EPs of the LUA holon executes moulding instances that its SP created.

#### 9.3.2.6 Moulding worker BASE plugins

Moulding workers offer services that they can offer to moulding activities. A moulding service needs to provide a LUA with four status updates as it performs the service:

1. Indicate when it starts the service;
2. indicate when the mould preparation is complete;
3. indicate when the moulding is complete; and
4. indicate when the mould has been vacuum bagged.

Additionally, the resource offering the moulding service can also inform the LUA of errors during the activity. Table 12 describes the plugin created for the BASE HRH-AS of the moulding workers to specialise the BASE HRH-AS in facilitating the business processes that the moulding workers are involved with.

**Table 12 – Moulding worker BASE plugins**

| Moulding Worker Plugin Type | Description |
|---|---|
| Scheduling Plugins | Receives RFPs for moulding requests and utilises Attributes to determine if the worker is qualified to perform the moulding request. If so, a moulding activity is placed on the Schedule. |
| Execution Plugins | Retrieves moulding activities from the Schedule and requests a moulding interface service from the moulding interface holon. The EPs use the data sent by the interface holon to receive activity state information from the worker. This state information is used to send the state updates to the LUA. |
| Reflection Plugins | Picks up completed moulding activities on the Bench and biographs them. Also receives review data on completed parts from quality inspection services further down the line. |
| Analysis Plugins | Updates the Attributes with personal Standard Work times on activities performed and keeps count of the different types of mouldings done. |

Figure 53 shows a tablet that was served an interface session by the interface holon. This instance has a BASE Service Provision Contract with the worker’s administration shell.

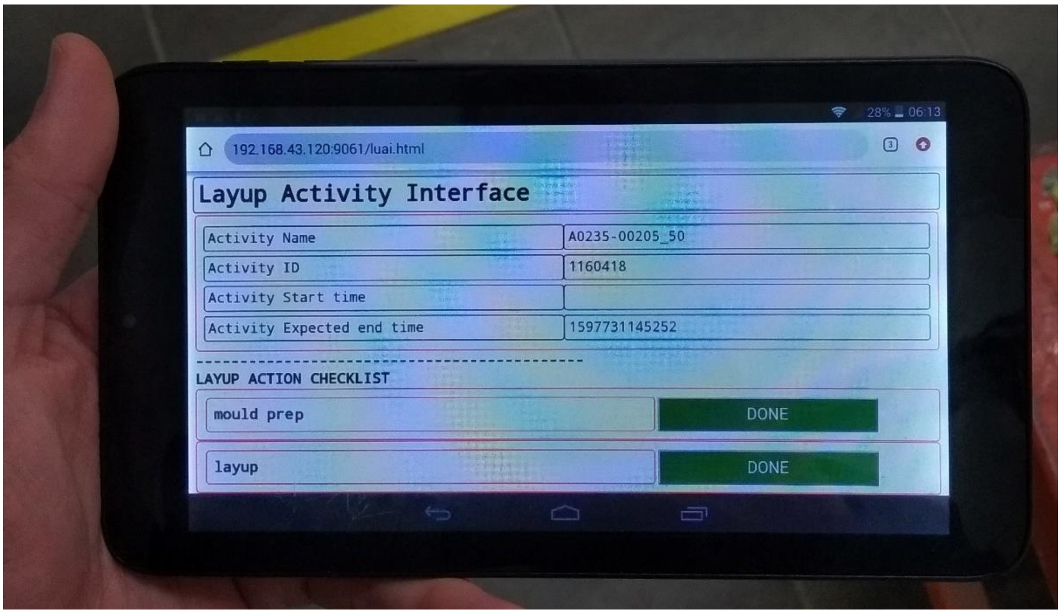



**Figure 53 – Moulding interface on a tablet, serving a subscription by a worker's BASE Execution Plugins**

#### 9.3.2.7 Line lead BASE plugins

Table 13 shows the line lead's BASE plugins that were developed, along with a description of their roles.

**Table 13 – Line lead BASE plugins**

| Line leader Plugin Type | Description |
|---|---|
| Scheduling Plugins | Receives RFPs for providing a supervision service on moulding activities. Supervision activities are placed on the Schedule. |
| Execution Plugins | Picks up supervision activities and subscribes to a Line lead interface service if it does not already have a running contract. Any new supervision activities and their details are sent to the interface service instance to which the Line lead is subscribed to. Activities that are being supervised can send status updates to the EPs which will display the necessary information through the subscribed interface service. |
| Reflection Plugins | Retrieves completed supervision activities and biographs them. |
| Analysis Plugins | No APs were developed for the Line lead in this case study but can be included in the future to support any specified analysis. |

Figure 54 shows a moulding supervision interface instance on the Line lead's tablet. Picture A of Figure 54 shows three activities being supervised, along with their expected start and end times on the day's timeline. Picture B of Figure 54 shows that one activity was started (indicated by changing to green) and selecting the activity reveals the worker currently subscribed as the moulder. This image was produced for the industry partner who used the abbreviation STDW for Standard Work.

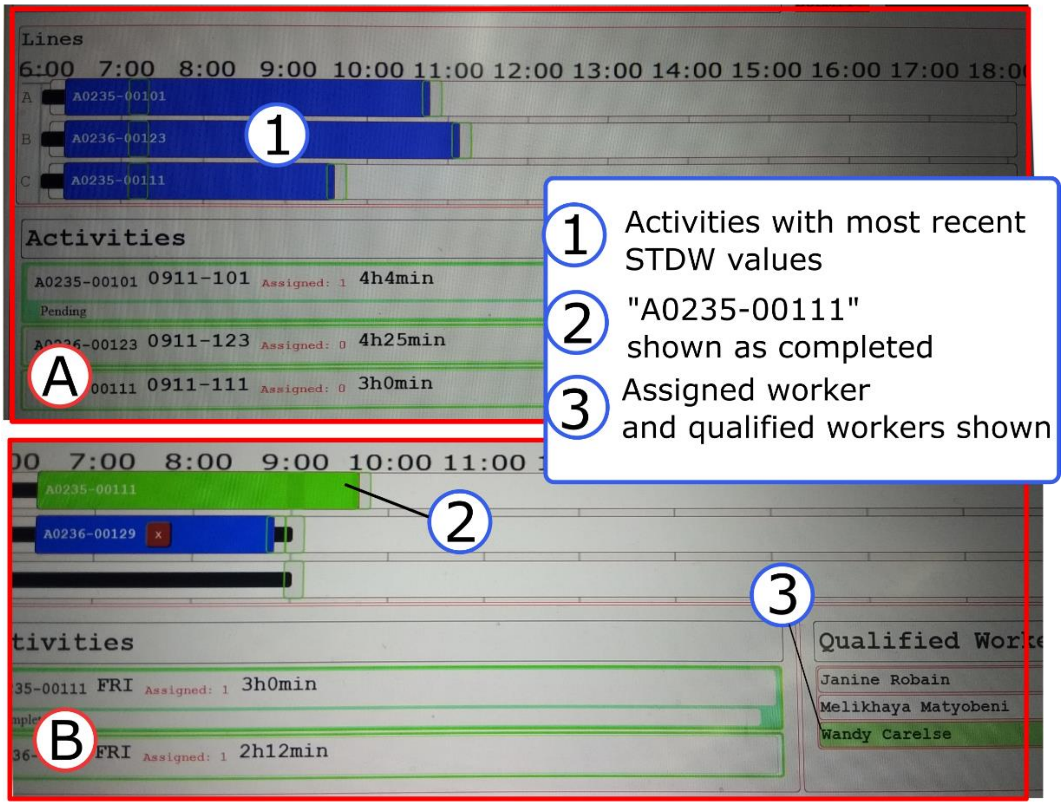



**Figure 54 – Line leader activity monitoring interface**

9.3.2.8 Scenario 1 holarchy overview

Figure 55 shows the holarchy involved with the moulding activity. The LUA acts as the orchestrating holon between the resources it needs. The HRH-AS receive and deliver data meant for the human through interface services they subscribe to.

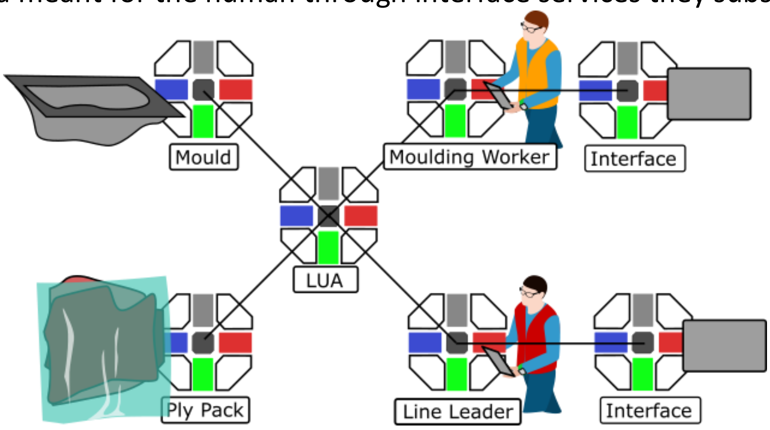



**Figure 55 – ALPs involved in error reporting with BASE**

The LUA instance is fed status information about the activity from the mould, worker, and ply pack ASs. All activity events are recorded as part of the execution data block of the 3SAL activity instance, while the activity state is also communicated with the Line lead's administration shell, which will utilise interfacing services to communicate this data to the Line lead.

9.3.2.9 Scenario 2 with BASE enabled assets

The business processes for which the manual ALP processes were observed were then replaced with appropriate BASE administration shells for resources involved along with plugins specific to the role they play in these processes. The following headings discuss how these were implemented and demonstrate the data they produced.

Providing feedback on completed activities

Moulding workers are resource holons contracted by the LUA type holon they deliver their services too. The LUA holds record of the worker, tools, and material resources involved. The activity produces a moulding instance resource that will form part of other holarchies further down the production line. Each activity that leads up to the final product can feed information back to previous activities which the previous activities will handle through their reflection plugins. A worker will receive feedback through the reflection plugins on his administration shell, as reported by quality inspection activities downstream.

Providing access to historical production information

Figure 56 shows an infographic constructed from the data in the LUA holon's Biography. Of the thirty activities involved, a breakdown of the errors reported is illustrated. Each activity entry has the structure shown by Table 14.

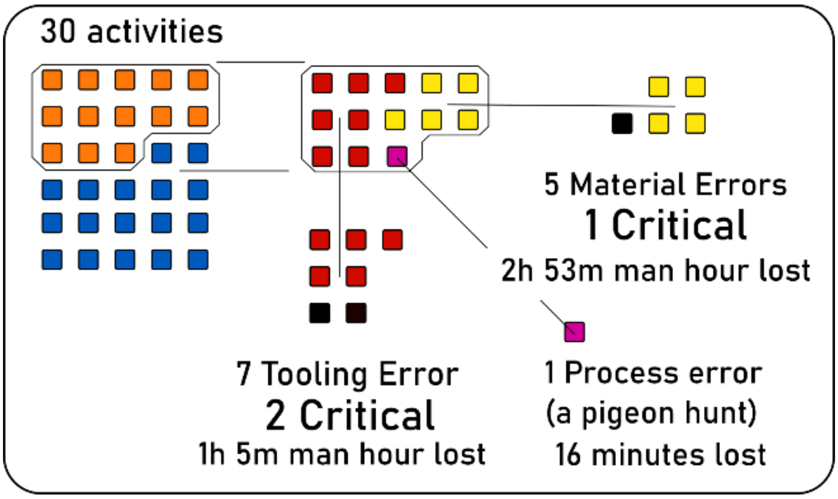



**Figure 56 – Moulding activity outcomes illustrated.**

**Table 14 – A Biographed 3SAL activity in the moulding activity BASE shell**

| Activity ID | Activity Type | Schedule Time | Start Time | End Time |
|---|---|---|---|---|
| A0235-00114_01-475001_C-295 | A0235-00114 | 1599542616361 | 1599542655174 | 1599549316157 |
| Schedule Data | Description: <<part description>><br>std_w : <<Most recent Standard Work data>><br>steps : <<List of process steps for activity>><br>Line leader: << Details of the line leader assigned>><br>Required resources : <<List of required resources for this activity>> | | | |
| Execution Data | Outcome: “success” \| “failed”<br>Activity Steps: <<list of activity steps and their recorded durations>><br>Error Reports: <<List of error reports generated for this activity>> | | | |
| Post-execution data | POTENTIAL QUALITY REVIEWS<br>POTENTIAL CUSTOMER REVIEWS<br>ETC… | | | |

<u>Obtaining Standard Work data on activities</u>

The worker’s BASE HRH-AS EPs, which manage their involvement in moulding activities, record their inputs through the moulding interface. This allows per-activity time analysis of not only the activity duration, but also the involved actions of mould preparation, layup and bagging. The acquisition of this information requires no additional ALPs and is stored as part of the 3SAL activity instance.

Due to the amount of activities that are automatically captured, statistical analysis is possible to determine standard deviation of activity durations. The industry partner reported that standard deviation was a valuable indicator of process maturity; however, it could not be determined previously due to the AL effort it would have required.

The LUA holon APs uses all biographed activity instances to update its Standard Work value in its Attributes. Figure 57 is an infographic produced for the industry partner as a detailed breakdown of the first production run of part “205”. Figure 58 illustrates the time analysis data retrieved from the layup activity holon’s Attributes. This data was produced by the layup activity’s APs. Standard deviation is a statistical value that depends on many data points, for demonstration only activities with three or more data points had their standard deviation determined to demonstrate the capability.

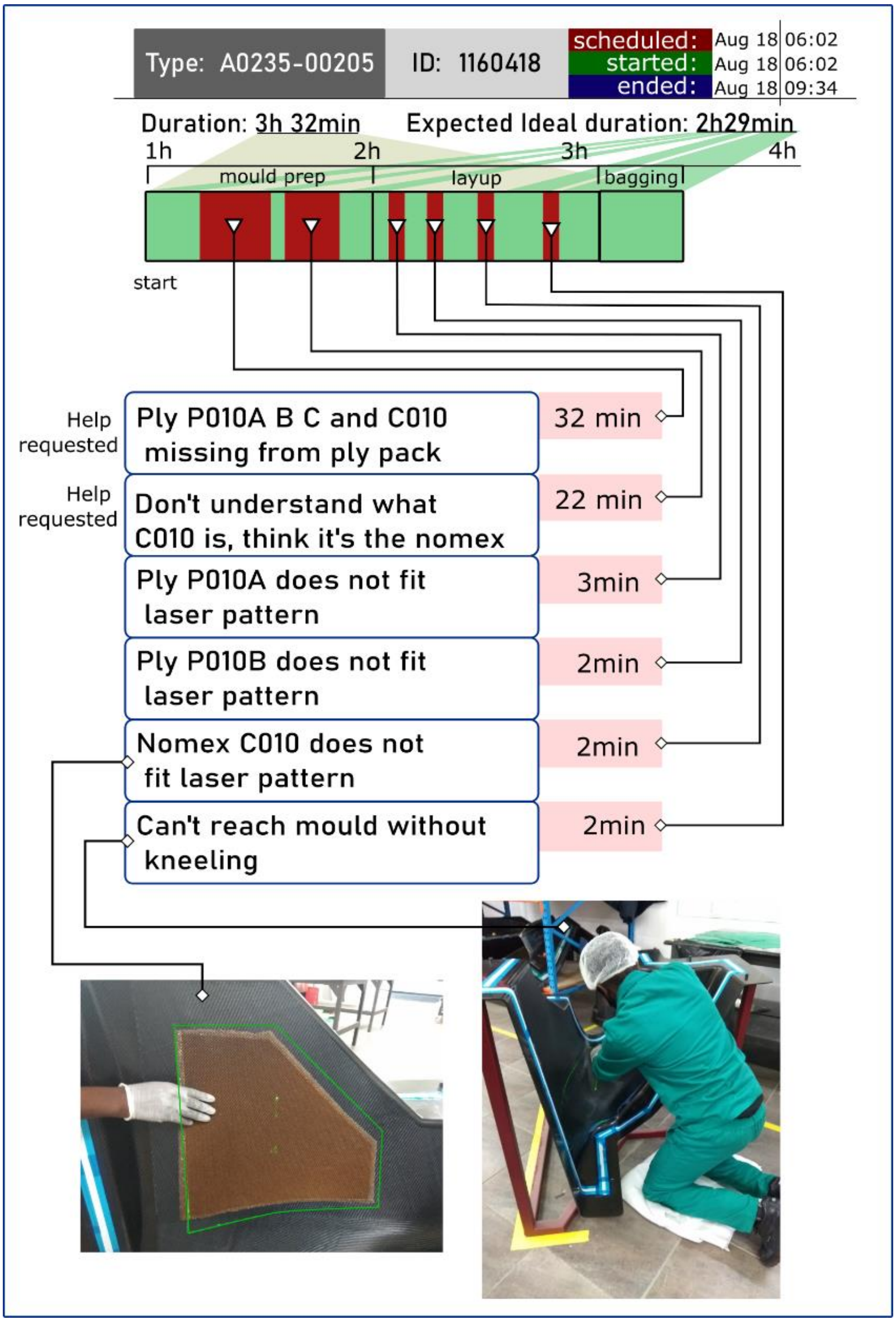



**Figure 57 – Infographic produced for the industry partner on first run of part "205" which did not yet have execution information**

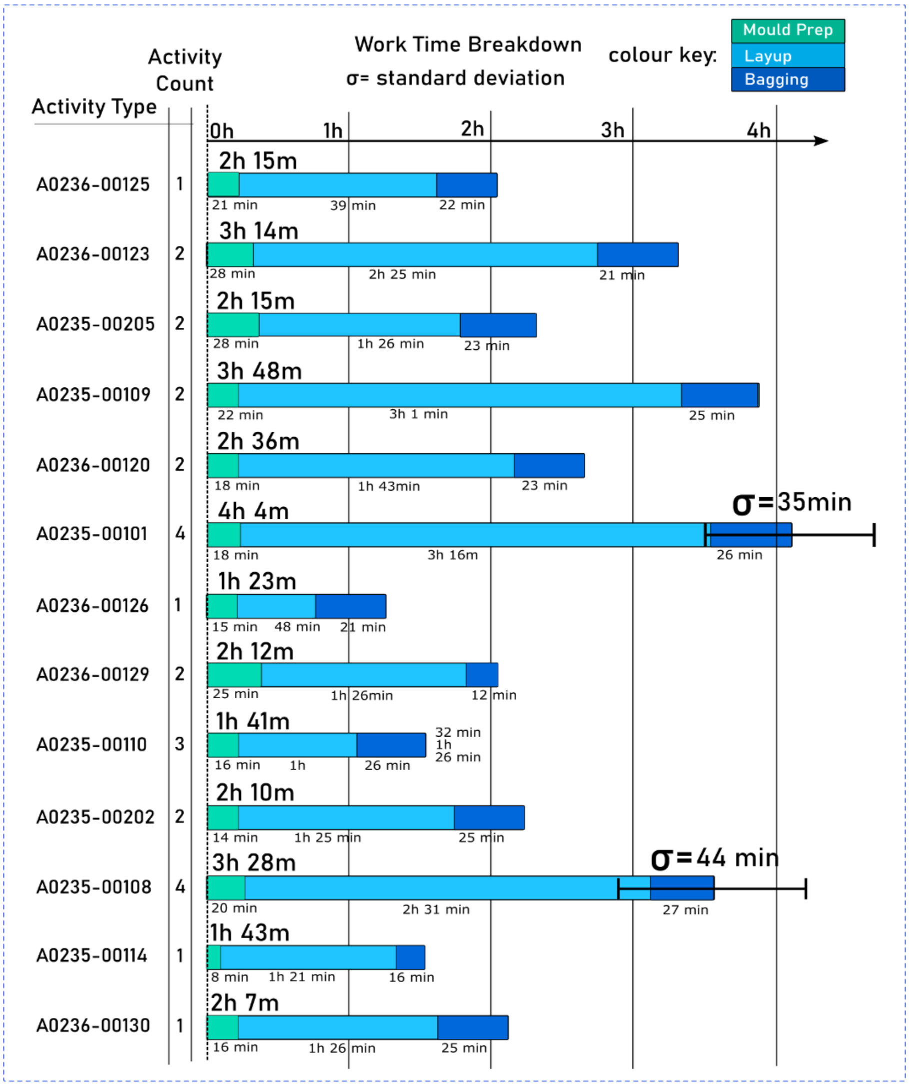



**Figure 58 – Standard Work for all activity types observed**

<u>Tooling and maintenance</u>

Despite the best efforts of workers to use the mould usage tracking system correctly, the industry partner reported that they often encountered problems with the system. The AL effort it takes of workers to manage this data forms part of the cause to these problems.

During the second week of the case study, a BASE administration shell was created to represent two of the commonly used moulds. Simple plugins were developed to keep track of their usage and indicate how many uses they have before they are required for cleaning. Since the mould's administration shell could be contacted by the Line lead's administration shell, the mould status became a live feed to the Line lead through his interface. This reduced the man-hours required for any ALPs involving mould information to zero.

# 10 Evaluation

This section discusses the observations made in Chapter 9 during the case study implementations and evaluates the ALPs involved in the business processes that formed part of the case study. This chapter will start with the evaluation criteria, describing a method for measuring the effectiveness of an ALP by defining characteristics of its component functions. The different mechanisms observed to perform the AL functions of these ALPs are then identified. These mechanisms are then evaluated according to their effectiveness and compared with each other.

## 10.1 Evaluation criteria

### 10.1.1 **Methodology**

The evaluation criteria formulate a set of metrics by which the effectiveness of ALPs can be determined. The methodology identifies the mechanisms used to fulfil various AL Functions (ALFs) that an ALP consists of, and then compares the combined effectiveness of these mechanisms between different ALPs for a business process.

The ALP comparison will be used to evaluate the effectiveness of the BASE architecture at addressing the problem of human integration, which was determined to be an issue of effective AL. It should be noted that the criteria developed here does not aim to evaluate an organisations decision making and business processes, and only evaluates the effectiveness of the AL involved in these processes.

As described in Chapter 3, an ALP is the process of data exchange between two influential entities in an organisation. A simple measure of an ALP's effectiveness can be done based simply on the time the ALP takes to complete; however, as this section will show, this only reveals a fraction of what makes an ALP effective.

Considering ALP effectiveness as a time measurement, the metric of *data-to-decision time* is presented. Figures 10 and 11 in Chapter 3 illustrate this concept. When a decision has been made by an entity, it will result either action taken or an influence on other entities. If the receiving entity is influential (i.e. can make decisions itself) a time can be measured from the instantiation of the initial decision to the time it is possible for the receiving entity to start making its own decisions. Minimising data-to-decision time is critical to an organisations effectiveness.

Although minimising data-to-decision time is an easily understandable measure, it only covers a small portion of what makes an ALP effective. Aspects of data degradation, the ability to locate the right source or destination for the data, and the ability of two parties to understand one-another when exchanging data are some examples of AL characteristics which influence effectiveness. Section 10.1.3 describes the characteristics of effective ALPs in more detail.

An ALP can consist of a complex series of mechanisms that each contribute to the completion of the ALP. For example, consider the ALP to notify workers of a schedule change on the shop floor. The goal is to deliver this information to the workers and the process can consist either of typing and sending an email, walking to a notice board and writing on it, walking to the workers and telling them, or any number of other processes. The mechanisms used in the example serve to fulfil one of four identified AL functions that all ALPs consist of. To evaluate different ALPs that serve the same purpose they need to be broken down into their component functions – at which stage the mechanisms that fulfil those functions can be evaluated. The four identified AL functions are discussed in the next section.

### 10.1.2 **Key Administrative Logistics Functions**

ALPs between two organisational entities consist of a combination of the following four AL functions:

- Data storage
- Data source/destination identification
- Data formatting and translation
- Data transport

Speaking, sending emails, writing on a notice board, and all other interactions between organisational entities are examples of mechanisms that fulfil a combination of these four functions. Descriptions of these functions and what they entail are given in Section 10.1.4.

### 10.1.3 **Characteristics of effective Administrative Logistics functions**

The effectiveness of these four AL Functions (ALFs) are determined by the three characteristics of *responsiveness, accuracy* and *visibility*. The following paragraphs describe these characteristics in more detail.

Delays, lag, latency and other problems that adversely affect how quickly the correct data reaches its intended destination will have negative consequences on decision making. The recent surge about “real-time” abilities of dashboards and other factory management systems aim to improve responsiveness of AL.

Accuracy measures the effect that data degradation, mistranslation, misinterpretation, source accuracy or routing problems has on decision making and decision execution.

Visibility consists of two components, namely *availability* and *traceability*. Availability of the AL function is determined by the amount of times it was needed over the times it was available for use. Traceability of a mechanism is determined by the ability to retrieve records about which entities were involved, what message paths were used, when an AL mechanism was used, etc. The widespread adoption of version control systems (e.g. Git[5]) is due to the excellent traceability it provides and the reason it is hosted on cloud platforms gives it excellent availability.

The effectiveness of a mechanism that performs an ALF can be determined by measuring its performance according to these three characteristics. Metrics for these characteristics are described in Section 10.1.4.

### 10.1.4 **Metrics for Administrative Logistic mechanism effectiveness**

This section will discuss how AL mechanism effectiveness can be measured through the characteristics of responsiveness, accuracy, and visibility. Each characteristic is measured with a set of metrics, which can be used as a basis for comparison between different mechanisms.

#### 10.1.4.1 Data storage

Data storage can be thought of as a form of data transport between decision makers not over space but over time. Human memory, hard drives, RAM, paper files and notice boards are all forms of data storage. These storage methods differ vastly in their three characteristics of effectiveness. Trade-offs between the three characteristics are determined by the needs of the application. As an example, a notice board may have good responsiveness and visibility to humans, but may lack detail, while a sensor value of a machine in RAM has excellent agility and accuracy, but very poor visibility.

Data storage responsiveness is measured through properties listed in Table 15. The time it takes to obtain the required data in the storage includes latency, access time, or waiting time if the data source needs to deal with multiple requests. The rate at which data can be transferred determines the limit to the amount of data that can be retrieved over time.

**Table 15 – Storage Responsiveness**

| Criteria | Property | Unit |
|---|---|---|
| Responsiveness | Time to obtain/store data | seconds |
| | Database bandwidth | (Mb/s) |

[5] Git is a distributed version control system for tracking changes in files

Storage accuracy can be measured by four metrics: the degradation of data over time, the compression when data is stored, the storage size limit and the data recentness – as listed in Table 16.

**Table 16 – Storage Accuracy**

| Criteria | Property | Unit/Rating |
|---|---|---|
| Accuracy | Data degradation | %/hr |
| | Data compression | % |
| | Data recentness | X/3 |
| | Data size limit | Mb |

The recentness of a data source is a measure of how up-to-date the data is to the truth. The industry partner's staff reported this property to be one of their largest concerns, and from their descriptions the rating in Table 17 was created.

**Table 17 – Data storage recentness**

| Rating | Description |
|---|---|
| 0/3 | Not reliable at all |
| 1/3 | Only updated when noticed |
| 2/3 | Updated irregularly with new data |
| 3/3 | In sync with newest data |

Storage visibility can be measured by its availability, and the traceability of the read/write requests. Table 18 contains the metrics that describe storage visibility.

**Table 18 – Data storage visibility**

| Criteria | Property | Unit/Rating |
|---|---|---|
| Visibility | Availability of the mechanism | % |
| | How traceable are the write/read requests? | X/3 |

Traceability of read/write requests are measured as described by the score in Table 19. This score was created by the description of staff at the case study partner. Their experiences with finding data for their decisions yielded these four categories. This traceability score will be used to evaluate other ALF mechanisms in a similar manner.

**Table 19 – AL traceability score**

| Traceability | Description |
|---|---|
| 0/3 | Not traceable at all, no data on its existence |
| 1/3 | It is possible to see what happened |
| 2/3 | It is possible to see when it happened and what happened |
| 3/3 | Timestamps, parties involved, and an ID of the action |

10.1.4.2 Locating a data source or destination

Data can be stored in an ideal location, advice and information may exist, or an instruction may be ready to be given, but the right location for the source or the destination of this data needs to be found. In many instances the problem of locating a source or destination may be mitigated by making the knowledge a fixed part of an organisation's structure (e.g. always report fires to this person or always write your complaints on the form in this room), but when organisational structure needs to change, or an unforeseen event occurs, these channels need to be newly formed.

If the data destination is a person, the location of that person could change, or they could be absent. In any case, the function of locating the correct data source or destination is a critical function to AL.

The characteristics for the ALF of locating sources or destinations will be presented in a single table, Table 20. The responsiveness of locating sources or destinations can be measured by the time it takes to find the right sources/destinations. Accuracy of locating sources/destinations can be measured by the success rate of finding the right destination or source. The visibility of locating sources/destinations can be measured by the traceability of the search and the availability of the mechanism used.

**Table 20 – Locating data source/destination visibility.**

| Criteria | Property | Unit/score |
|---|---|---|
| Responsiveness | Time to connect sources/destinations | s |
| Accuracy | Success rate | % |
| Visibility | What is the availability of the mechanism? | % |
| | What is the traceability of the mechanism? | X/3 |

10.1.4.3 Transporting data

If a decision was made, whether it be a high-level management directive or a smoke alarm detecting a fire, the results of that decision need to be transported to the right influential entity through the most effective path. Furthermore, this transport must preserve data accuracy and occur with the shortest possible delivery time. Transporting data serves the same purpose whether it is over a network between two machines, an email or word of mouth.

The responsiveness of a transport mechanism can be measured by the time it takes the data to reach its destination and the bandwidth limits of the mechanism. The accuracy of transport can be measured by the degradation of the data, the compression of the data and the percentage of data that reach the intended destination. The visibility of data transport can be measured by the availability of the transport mechanism and the traceability of the data transport path to the destination. Table 21 lists the metrics for measuring the effectiveness of mechanisms performing the data transport AL function.

**Table 21 – Data transport characteristics**

| Criteria | Property | Unit/score |
|---|---|---|
| Responsiveness | Bandwidth limit | MB/s |
| | Time to reach destination | s |
| Accuracy | Message degradation | % loss/minute |
| | Data compression | % |
| | Correct destination | % |
| Visibility | Mechanism availability | % |
| | How traceable is the message path | X/3 |

10.1.4.4 Transforming data

When data is given from one component to be consumed by another, the data may need to be translated, formatted, or changed in some way that suits the destination. This transformation of data is often due to differences in language, protocols, software version or skill levels between the communicating entities. A simple example may be formatting data from XML into JSON structures. A more challenging example would be translating G-code into a form that is understandable by humans, such as a path or shape.

It is expected that interpretation and context may significantly affect the way humans consume data. As such, effectiveness of the data transformation ALF through qualitative metrics would likely be of more value when the entities involved are humans. However, some quantitative metrics can be developed to provide insight into the performance of data transformation functions by different mechanisms.

The responsiveness of a transformation can be measured by the time the transformation takes. The accuracy of a transformation may be subjective when it comes to humans and can be a complex property to measure, as such it is left as percentage representing the amount of the message that came across correctly to the consuming entity. The visibility of transformation can be measured by the availability of the transforming mechanism, as well as the traceability of the transformation in terms of sources, transformation method, and the entities involved. Transformation characteristics are listed in Table 22.

**Table 22 – Transformation characteristics**

| Criteria | Property | Unit/score |
|---|---|---|
| Responsiveness | Time to formatting/translation | s |
| Accuracy | Translation accuracy | % |
| Visibility | Availability of transformer | % |
| | Traceability of the transformation used | X/3 |

## 10.2 Identification of Administrative Logistics Functions and Mechanisms in the Case Study

This section identifies the AL functions that make up the observed AL processes in the case study of Section 9.3. Since many of the ALPs in this case study were manual, the cost of an ALP is measured in man hours. The man hours witnessed is given as part of the illustrations in this section and will be discussed in the evaluation section.

### 10.2.1 Scenario 1 – Monitoring Moulding Activities

During scenario one, several ALP's were witnessed that consisted of various mechanisms for performing AL functions they were composed of. A generalisation of the observed AL functions and mechanisms is presented in Table 23, from which the ALPs in this scenario are reconstructed and analysed.

**Table 23 – Observed AL functions and mechanisms during activity monitoring**

| Description | AL function | AL mechanism |
|---|---|---|
| Walking to go look at information at source or storage | Transport | Walk-to-see |
| Information stored in tables or notice boards on a computer or on a paper. | Storage | Noticeboard / Spreadsheet |
| Information being remembered by a person to tell someone else, make decisions later, or record to other storage. | Storage | Memory |
| Recorded information on a Digital Messaging Platform (DMP) that can be looked up later | Storage | Digital Messaging Platform |
| Sending information to other people via a WhatsApp or email | Transport | Digital Messaging Platform |

The observed ALPs for obtaining an activity's state can be summarized as the three paths shown in Figure 59. ALP_1.1 shows a walk-to-see mechanism for performing the transport ALF, along with a spreadsheet and human memory storage mechanism used. An alternative option, indicated as ALP_1.2, involves a DMP (WhatsApp). The third path, ALP_1.3, is supported by the implementation of the BASE architecture. It should be noted that neither of the first two processes involved a data locating mechanism, as this information is already known as part of training.

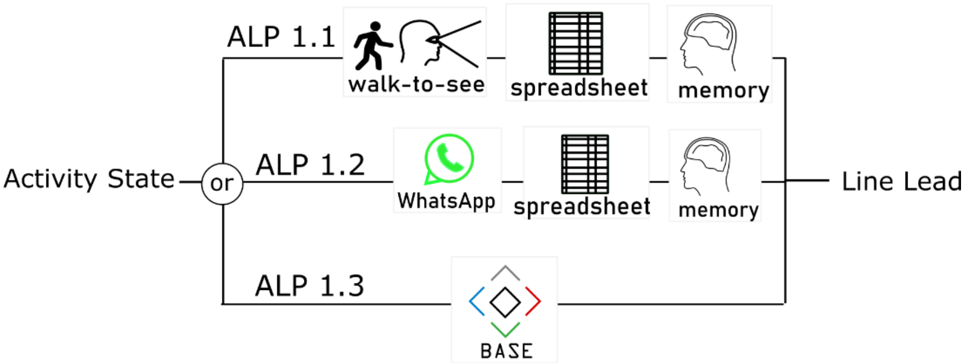



**Figure 59 – Line lead ALPs to obtain activity states**

The Line lead can be notified of information that a worker noticed through the three ALPs shown in Figure 60. Indicated as ALP_2.1, the worker can remember what they wanted to say and walk to find the Line lead to tell him, ending with the line lead's memory. Walking to tell often involves walking to find followed by speech – the mechanism used has equivalent characteristic to walk-to-see apart from the bandwidth of information transferable. The worker can also use WhatsApp to message the Line lead (ALP_2.2), or the worker can use his BASE administration shell and its subscribed interfaces. Again, neither of the first two ALPs involved a locating mechanism, as this information is already known as part of training.

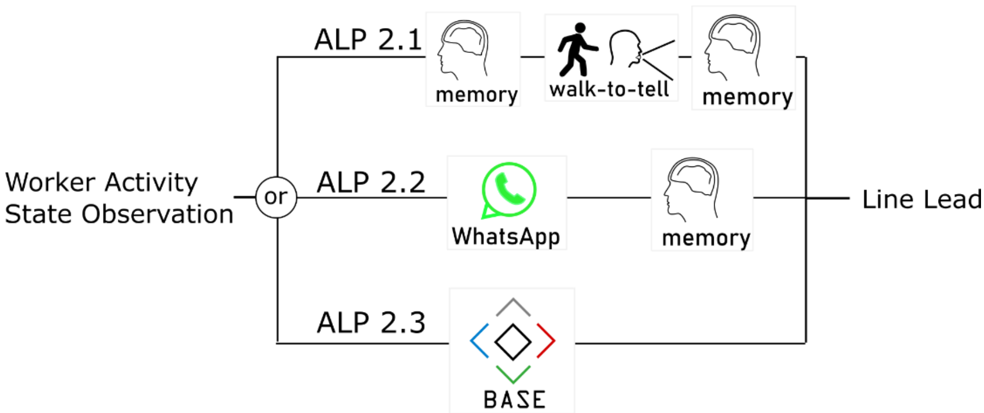



**Figure 60 – Worker ALPs to report activity errors or state**

### 10.2.2 Scenario 2 – Quality improvement and process visibility

This section illustrates the AL mechanisms involved in the observed ALPs of scenario 2. As in scenario 1, the AL functions comprising the ALPs were generalised and presented in Table 24. From Table 24 the ALPs are reconstructed for evaluation.

**Table 24 – Observed AL functions for scenario 2**

| Description | AL function | AL mechanism |
|---|---|---|
| Walking to go look at the outcome of their mouldings | Transport | Walk-to-see |
| Walking to tell a worker about the outcome of their mouldings | Transport | Walk-to-tell |
| Manually recording Standard Work times | Transport | Sit and watch |
| Storing Standard Work times in a spreadsheet | Storage | Noticeboard/spread sheet |
| Finding Standard Work values for activities | Transport | Walk-to-see |
| Manually record mould usage | Transport | Walk and write equivalent to Walk-to-see |
| Storing mould usage information | Storage | Noticeboard/spread sheet |
| Finding mould usage information | Transport | Walk-to-see |
| Checking material out life | Transport | Walk-to-see |

#### 10.2.2.1 Obtaining Standard Work data on activities

To obtain Standard Work, two ALPs were possible. In ALP_3.1, an industrial engineer needs to observe the layup activities throughout their entire duration and record the times manually. ALP_3.1 is unique and requires performing a time study to sit and watch activities being performed while recording observations on paper and later transferring the recordings to an Excel spreadsheet. ALP_3.2 uses the BASE administration shell execution plugins to record the data of activities as they progress. These ALPs are illustrated in Figure 61.

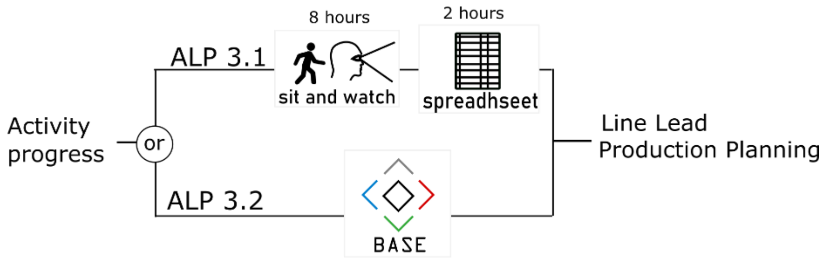



**Figure 61 – Observed ALPs for obtaining Standard Work information**

10.2.2.2 Tooling maintenance

To observe a mould or tool state, the Line lead needs to walk to the location of the mould and read its spreadsheet. He must remember this information to make use of it later on. He can also send a WhatsApp message to the moulding shop and wait for a reply. The third ALP available is to use his BASE administration shell to contact the admin shell of the mould. The mould's administration shell can also push information to the Line lead if its plugin components were written to react on certain conditions. These ALPs are illustrated in Figure 62 along with their component mechanisms.

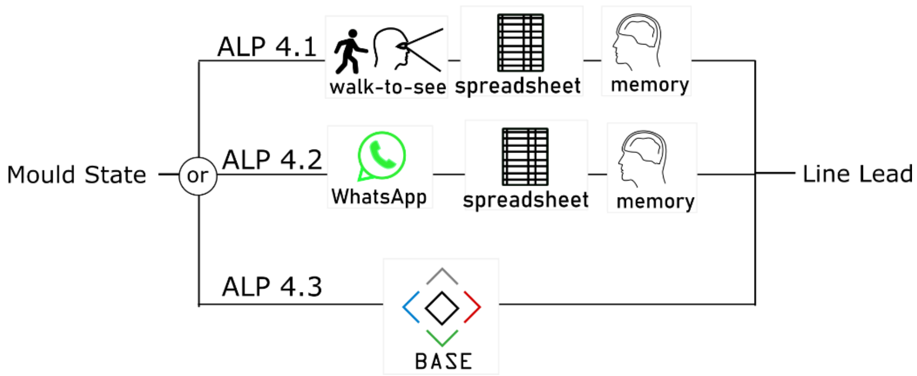



**Figure 62 – ALPs observed for obtaining tooling information**

10.2.2.3 Worker feedback and quality improvement

For a worker to receive quality feedback, advice, or praise on the outcome of parts they completed, the worker could either walk to see the outcome at the quality assurance station (ALP_5.1) or someone could walk to tell them (ALP_5.2). These ALPs including the ALP using the BASE HRH-AS (ALP_5.3) is illustrated in Figure 63.

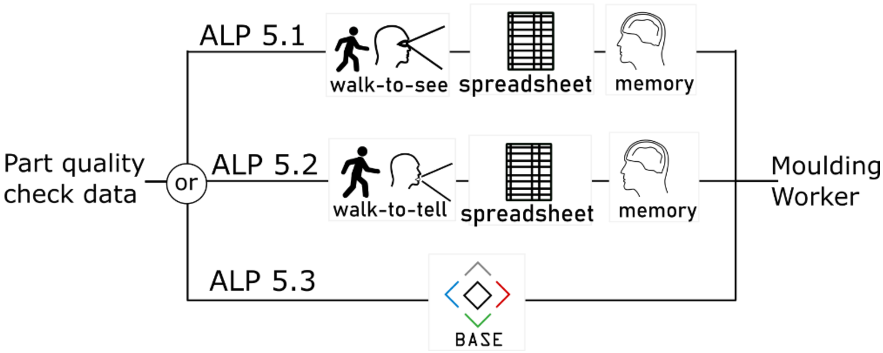



**Figure 63 – ALPs observed for worker feedback and quality improvement**

## 10.3 Evaluation of Administrative Logistics Mechanisms

The identified AL mechanisms of Section 10.2 need to be evaluated according to their effectiveness in performing the AL functions. This will allow the ALP that consist of them to be evaluated by the combined effectiveness of their component mechanisms. The ALPs for each scenario can then be compared.

### 10.3.1 **Data storage**

The following mechanisms were observed to perform data storage during the case study. Each data storage mechanism will be described before a final summary of their characteristics is given in a table.

#### 10.3.1.1 Human memory

It is not possible to summarize the abilities of human memory by the few metrics described here. There are several major advantages to the human mind that mitigates the score given here, such as contextual understanding, pattern recognition, and imagination. When trying to measure the degree of raw data compression in the mind, Chaitin (2006) stated that “compression is comprehension” – meaning that the human mind is not just a storage mechanism, but can immediately improve the value of data it is holding, depending on the human’s understanding and contextual awareness. However, these advantages only apply when this data is shared with other humans.

Considering the value of human to human communication, there is often data that workers need to report for which they lack contextual understanding. This contextual understanding is critical when reporting raw data (e.g. images, numbers, geometry, etc.) that the engineering, project planning or quality assurance departments need to interpret. It is often in these cases where human memory and summaries fail.

The metrics used to evaluate storage do show limitations to human memory, when numbers, times, images, or geometric data needs to be accurately recalled. A camera picture is better than a hand sketch from memory, and even if the human can imagine what they saw clearly, the recollection of this data is limited to drawing ability. Therefore, human memory suffers from data compression loss, data size limits and bandwidth.

Memory, as considered in this study, does not take into account episodic recollection, but rather short-term recollection of details, such as part numbers or the placement of tools. Short term memory is limited to 7 ± 2 items or concepts, and only three that can be worked with at a time. Once concentration is lost, the chance of recollection is seriously diminished (Sanders & McCormick, 1993). Long term memory is also subject to deviation over time. Not considering data degradation, and assuming data "chunking" is used (Thalmann, Souza & Oberauer, 2019), human memory is limited to only around 33 bytes (assuming the concepts remembered are words or numbers).

Figure 64 illustrates the recollection ability of humans in percentage against the elapsed time. It shows how even after 10 minutes the ability to accurately recall an event or data is seriously diminished (Stahl, Davis, Kim, et al., 2010).

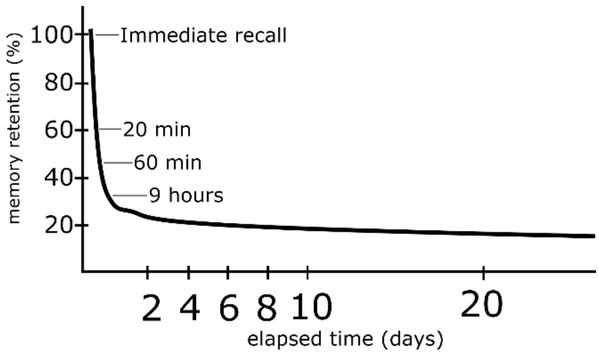



**Figure 64 – The forgetting curve, recreated from (Stahl et al., 2010).**

Taking a 40% recollection as the limit of use, a linearized estimate for data degradation is made from Figure 64 as 2% per minute, effectively lost after a day. Therefore, human memory will only be considered an available AL storage function for short term ALPs.

The responsiveness of short term memory is in the order of seconds, with a reported data stream of about 1.6 words spoken per second, with each word consisting of on average 4.7 characters (Norvig, 2012). In data terms, that is approximately 7.52 bytes per second or roughly 8 bytes/s. While speaking is not the only form of communication, humans can't use more than one channel of communication at a time – leaving this as the bandwidth limit.

Human memory is not always an up to date single source of truth on a piece of information, and relies on the human being given new information, or going to check if new information is available. Therefore, it will be rated as updated irregularly with new data for a recentness score of 2 out of 3, as per Table 16.

The size limit of human memory is complicated by the ability of humans to comprehend and summarize data. These benefits do not reflect when the data needs to be communicated to a digital system and is then limited to what can be expressed in 7 concepts – effectively 7 words or 33 bytes (7 words of an average of 4.7 characters). A person can only give information to one receiver at a time, or when broadcasting/shouting, the message needs to be applicable to all receiving parties. This can also only be done when not busy with another task. This, combined with the fact that the information can only be obtained in the physical presence of the human or through actively messaging over a DMP, gives the availability of this data a low score.

The availability of a worker or the line lead's memory is determined by the amount of time the person is available to be questioned over the total time the information may be needed. This time was observed to be around 10 minutes between moulding activities where workers could look at their phones, and 20 minutes during tea breaks. Putting the total available time to answer questions conservatively at around 30 minutes of the 9 hours from the perspective of the line lead (roughly 6% availability).

It was also observed that the line lead would roughly get two chances to visit the moulding shop per day, since it costs him around 25 minutes of his time on average to do this. Subsequently, another conservative estimate can be set to 40 minutes out of the total 8 hours of availability to query workers directly. This totals to an availability of moulding worker memory-based data of roughly 14%.

#### 10.3.1.2 Excel sheets and notice boards

Excel spreadsheets and notice boards were observed as methods to recall worker skills, required orders and tool states. The average time to find a required piece of data on these sheets were 20 seconds (averaging observations from short and long lists). Since notice boards can contain images or graphs, this increases the available bandwidth to the limit of the reader's perception and understanding of the information source.

Data on a spreadsheet or noticeboard does not degrade, nor will it be assumed that data needs to be compressed (although this may be a conservative assumption) – where these methods suffer is in the data recentness. Observing the use of these methods showed that the data is only updated if there is a person responsible and aware that a spreadsheet is not up to date. Often this leads to sheets only updated when it is noticed and there is time for someone to do it.

Performing the update is not critical in many cases, since workers will simply rely on their experience of the process.

The irregularity of information updates does pose a problem when a process is changed or new parts are introduced. Spreadsheets or notice boards most often contain numeric or textual data but could contain graphs or some images; limiting their data size limit to a few megabytes. In most cases the author of the notice board or spreadsheet is known, but workers reading or updating the information are not recorded. As such, spread sheets or notice boards can only achieve a low traceability score of 1 out of 3.

The availability of a spreadsheet is high, since it does not rely on networks, power or attendance. However, spreadsheets can only be accessed by moving to their physical location – this will influence the availability of the mechanism, as is covered by the data transport ALF discussed in Section 10.3.3.

10.3.1.3 Digital Messaging Platforms

WhatsApp, email and Facebook are examples of DMPs that offer the advantage of real-time synchronization of data and easy accessibility to anyone with a phone or computer. For the sake of brevity, no differentiation will be done between the various DMPs, since typically they all support chat groups, dialog and similar media handling capabilities.

When data needs to be retrieved from a DMP, scrolling through a chat or finding an exact email chain either requires key words to be remembered or contextual information held by the parties during specific conversations. This contextual information often makes the message become more obscure over time (e.g. people will often talk about "that part" instead of "A0235-00122-345"). This, combined with the fact that there are often several concurrent conversations occurring in a group chat, makes retrievability of the data nearly impossible after a week according to the line lead. An exact measurement of the availability of data depends on the individual trying to find this data in the group chat history. The Line lead rarely attempts to look for data in the conversation more than two days back, and during the day would only look for part numbers or worker names and not for activity status or other data.

A distinction is thus made between short term (less than 2 days) and long-term availability. The short-term availability is considered to be 50%. For the above-mentioned reasons, long term availability is poor and even though the line lead reported long term DMP information is not used, the author believes a conservative estimate of 10% availability should be taken to factor in rare occurrences.

Furthermore, the time to obtain data in the short term is in the order of seconds, while long term data retrieval could take minutes. In the presented scenarios of scheduling and monitoring the short-term properties are used, while resource management and process improvement and visibility will use the long-term properties.

The data size limit for DMPs will be set to the capacity for photo sharing in WhatsApp, and it is taken that there is no appreciable data degradation with this mechanism. Finally, the data on a messaging application only updates when there is active conversation on the topic and, consequently, the mechanism's data recentness score is low at 1 out of 3 (as per Table 17).

Traceability and data degradation therefore suffer in the long run when using DMPs. DMPs can be noted for mitigating the spatial gap between decision makers, but does not address the temporal gap in long term ALPs.

#### 10.3.1.4 BASE data storage

An implementation of the BASE architecture uses digital storage, and all BASE activities are recorded in the 3SAL format. Therefore, activities managed through a BASE administration shell has no degradation and full traceability. Historic data is available through the Biography of any asset that was BASE enabled when it took part in an activity.

Although BASE is capable of streaming video, this was not tested – only photo capturing ability was tested and will be considered for assessing bandwidth capabilities, typically in the order of 5 MB/s. Being stored digitally, there is no data degradation. The data size limit is a hardware limit and will be placed at 10 gigabytes.

The availability of data in the implementation of BASE Administration shells is the same as Erlang's uptime reliability of nine nines (99.9999999%), meaning data is effectively always available between two BASE administration shells. A limitation exists when there are no plugins to interpret the data. For humans, this requires that their administration shell have interface plugins. Due to possible network instability or plugin and user interface bugs, the availability will be capped to 99%.

#### 10.3.1.5 Summary of data storage mechanism evaluation

Table 25 summarizes the evaluation of data storage mechanisms used in the two scenarios of case study 2. The AL processes for these scenarios are achieved through a combination of AL functions executed through these mechanisms.

**Table 25 – AL data storage function summary and comparison**

| Charact-eristic | Property | Spreadsheet | DMPs | Human memory | BASE |
|---|---|---|---|---|---|
| Responsiveness | Time to access data | ~20s | 20s (short)<br>10m (long) | ~ 4s | 0.5s |
| | Database bandwidth | ~1kb/s | 5 Mb/s | ~7.52b/s | 10 Mb/s |
| Accuracy | Data degradation | 0 %/min | 0 %/min | ~ 2%/min | 0 Mb/s |
| | Compression loss | 0% | ~10% | ~ 50% | ~10% |
| | Data recentness | 1/3 | 1/3 | 2/3 | 3/3 |
| Visibility | Data size limit | ~10 Mb | ~16 Mb | 7 concepts (~33 bytes) | 16+ Mb |
| | Availability | ~99% | ~50% (short)<br>~10% (long) | ~14%. | ~99% |
| | write/read request traceability | 1/3 | 1/3 | 1/3 | 3/3 |

### 10.3.2 **Locating data source or destination**

When a human needs to find the location of a source of data for a decision (which may be another person, a part or a folder), they often make use of other humans to direct them. In the case study, two mechanisms were identified to be commonly used on the shop floor. WhatsApp and walk-to-see. The BASE administration shell mechanism for finding information can only be considered if the resource in possession of the information is connected to the BASE holarchy with a BASE administration shell.

#### 10.3.2.1 Digital Messaging Platforms

In the case of WhatsApp being used to locate a data source or destination, various human factors come into play that affect the AL effectiveness. If the data source to be found is not on the application, but needs to be found by asking someone else, the availability of this channel is dependent on the response time of the other person. If the search is for a piece of data that was part of a conversation, the Line lead reported that it becomes impossible to find anything after a few days due to the growing number of messages in concurrent conversations. Time is thus a factor when considering the availability, and the scores will be broken into long term (more than two days) and short term (less than 2 days) retrieval.

10.3.2.2 Walk-to-see

The time it takes to find a source or sink by foot depends on the area that needs to be searched. The search area for information in the case study was observed to be between the Line lead's office, the kit cutting room and the moulding room, with a 200 m walkway between the buildings as illustrated in Figure 65. The total area to search was estimated to be around 1200 $m^2$.

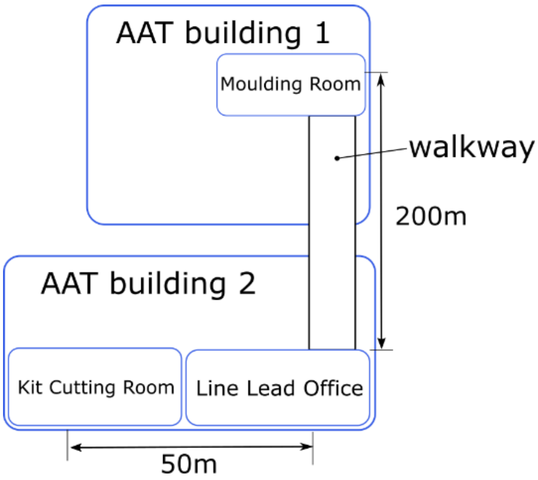



**Figure 65 – Industry partner moulding processes physical layout and buildings**

The observed average time for a worker to locate information about a particular ply-pack or tool during a process error was 25 minutes. The Line lead and other staff agreed that they would spend that amount of time looking for information if its location was unknown.

Out of a total of six times that a worker was noticed actively looking for information, four of the times they were successful while the other two times they came back and said they will try again later. This gives a 60% success rate for the walk-to-see mechanism. The Line lead reported that this correlated with their own experience.

10.3.2.3 BASE holon information search

When an activity or resource is BASE enabled, they are fully visible to all other holons. This means that not all data forms part of the reflection of reality that the administration shell tries to maintain, but this case study considers the most required and process-critical data involved. The LUA holon could form contracts with all required resources within 0.2 seconds, which means that finding a resource and the information its BASE administration shell has on it will be close to the same amount of time. The success rate will be the same as the system uptime reliability and if the decision formed part of an activity will have full traceability.

#### 10.3.2.4 Summary of data location mechanism evaluation

A summary of the evaluation of the three data location mechanisms observed in the ALPs of the case study is presented in Table 26.

**Table 26 – Comparing AL function effectiveness for locating sources or destinations**

| **Characteristic** | **Property** | **Walk-to-see** | **DMPs** | **BASE** |
|---|---|---|---|---|
| Responsiveness | Time to find sources or destinations | ~25 min | ~3 min | ~0.5 s |
| Accuracy | Success rate | ~60% | ~60% | ~99% |
| Visibility | Availability of the directory | ~30% | ~30% | ~99% |
| | Traceability for selecting a source/destination | 1/3 | 2/3 | 3/3 |

### 10.3.3 **Data transport**

Data transport is an AL function that is concerned with moving data from a source to a decision maker, between decision makers, or from a decision maker to a data store. Email, WhatsApp, and walk-to-see are three observed data transport mechanisms. For the sake of brevity, all DMPs (DMPs) are considered to have the same ALP characteristics, since they typically offer similar functionality.

#### 10.3.3.1 Digital Messaging Platforms

The availability of DMPs rely on various factors that depend on the humans using them. The availability of the DMP for transport is the same as its availability for storage and is dependent on the same factors. DMPs suffer no data degradation. In terms of data compression, WhatsApp clamps all images to a maximum of 800 by 600 pixels. This compression was not noticed to adversely affect decisions, and the compression loss will be taken as for JPEG images, reported at 10:1. However, it is considered that the perceptible loss does not affect the information that the image tries to communicate – leaving compression loss at less than 10% (Haines & Chuang, 1992).

#### 10.3.3.2 Walk-to-see

Consider the case where information destination is a human, but the source is fixed in space, such as a notice board or a component in a shop. The person needs to walk to the source of the data to observe it. Relative to the mind, it may as well be the source of the data that moves towards it. Walking to a data source is thus classified as data transport, and the total ALP availability depends on the availability of this transport mechanism. Having the time or the means to travel to the data source contributes to the availability of the data transport channel.

The line lead's responsibilities require him to obtain information from the entire manufacturing process and this only allows him time to walk-to-see a few critical points of data. The line lead was observed to have only around 30 minutes out of 8 hours to spend on finding information on the mouldings that were part of the case study. This leaves the availability of this transport mechanism very low at around 6%. This is only for obtaining data from this room in the factory, and it can be seen how this effect adds up when multiple pieces of information need to be obtained from different places.

Since the data is obtained from the source, this transport mechanism has no message degradation nor data compression. The line lead has access to all his senses once at the information source and, as such, the bandwidth limit will be considered limitless. The line lead may remember where he saw certain pieces of information, but the traceability decreases at the same rate as human memory, making it unreliable after a day. The traceability is thus awarded a score of 1 out of 3.

Walk-to-see is a powerful tool when complex problem solving, interpersonal relationships, or high clarity is required by the decision maker. However, it will be shown in the scenarios presented how this is excessive for many situations that simply require a Boolean, numeric or string type value. This "right sizing" of an AL mechanism will be discussed as part of the evaluation but does not form part of the effectiveness score.

Similar to walk-to-see is the mechanism of walk-to-tell which often includes the need of a walk-to-find mechanism to search for the person that the information is intended for. This mechanism will be treated to have the same characteristics in all regards to walk-to-see apart from the bandwidth limitation of human to human communication. If a worker needs to communicate by word of mouth it limits the bandwidth to 7.52 bytes per second or roughly 8 bytes/s.

#### 10.3.3.3 BASE data transport

Data transportation between BASE administration shells for this implementation took place through Erlang message passing. This means the transport availability is the same as the system uptime reliability – effectively always available. Time to deliver messages is in the order of milliseconds. The bandwidth limitation for BASE in this case study will be conservatively capped to the photo capturing capabilities of the user interfaces developed for use by the workers.

#### 10.3.3.4 Summary of data transport mechanism evaluation

Table 27 summarises the evaluation metrics of each observed AL transport mechanism.

**Table 27 – AL mechanisms for data transport effectiveness comparison**

| **Characteristic** | | Walk-to-see | DMPs | BASE |
|---|---|---|---|---|
| Responsiveness | Time to transport data | ~25 min | ~5 sec | ~2 s |
| | Bandwidth limit | Limitless | ~5 Mb/s | ~5 Mb/s |
| Accuracy | Message degradation | 0%/min | 0%/min | 0%/min |
| | Data compression | 0% | ~10% | ~10% |
| | Correct destination | ~99% | ~99% | ~99% |
| Visibility | Transport availability | ~6% | ~30% | 99% |
| | Transport traceability | 1/3 | 1/3 | 3/3 |

### 10.3.4 **Data translation**

Translation of data occurs when data is not in a form interpretable to the consumer or not compatible with the data store. In the case study conducted, it was deemed that no significant data translation or transformation occurred between sources and consumers, and if they did occur it was insignificant to the combined effects of all other ALPs involved. For simplification of the evaluation this AL function will not be included.

## 10.4 Comparison of Administrative Logistics Process effectiveness

This section will compare the ALP effectiveness in the scenarios described in Chapter 9. For the normal operations (i.e. without BASE enabled assets) it was determined that a combination of various ALPs were required in order to compare to the single ALP involving BASE components. This combination of observed ALPs was found to be a subjective process and, therefore, the most conservative methods of combination are used. For example, when a person can message the moulding WhatsApp group or Walk-to-see, the combination will assume both are done at the same time in all cases when in reality one AL process usually serves as a fall back on another.

The comparison is facilitated by first describing the method by which the effectiveness of AL functions is combined to represent the effectiveness of ALPs. On this basis, the effectiveness of ALPs, in the normal operation and BASE architecture implementation, is compared for the scenarios of the case study.

### 10.4.1 **Combining AL function effectiveness for total AL process effectiveness**

To compare the ALPs observed the performances of their component functions need to be combined. The combination of AL functions should consider if the functions are performed in series or in parallel. The combination of certain AL functions performed in series weakens their effectiveness. This is analogous to the "Chinese whispers" phenomenon where a message degrades as it passes through people in series.

In contrast, when the same message passes through two people in parallel, it comes across more accurately than if it was only passed on by one (as it allows for combined versions of their points of view). This means that the effectiveness of certain combined AL functions improves when performed in parallel. This effect is similar to that observed in the combination of springs. As such, the effectiveness of AL functions performed in parallel will be combined with the reduced sum operation (denoted by "∥"), which is the reciprocal value of the sum of reciprocal values.

In some cases, the effectiveness of an ALP will be limited by the lowest effectiveness value in its series of comprising functions. In other cases, the effectiveness metric is added, such as with metrics measuring time.

Table 28 defines how AL function characteristics can be combined when evaluating AL mechanisms that perform them to give an overall evaluation of the ALP. Some characteristics are unique to the AL function and cannot be combined, which is represented as a hyphen in the combination method column.

**Table 28 – Combination of AL characteristics**

| | **Admin Logistic Function** | | | |
|---|---|---|---|---|
| **Characteristic** | **Locate** | **Transport** | **Store** | **Combination method** |
| Responsiveness | Time to find sources or destination | Time to transport data | Time to retrieve or store data | Addition |
| | | Bandwidth limit | Bandwidth limit | Lowest value |
| Accuracy | | Message degradation | Data degradation | Reduced Sum (‖) |
| | | Message compression | Data compression | Reduced Sum (‖) |
| | Success rate | | | - |
| | | Correct destination | | - |
| | | | Data recentness | - |
| | | Message size limit | Entry size limit | Lowest value |
| Visibility | Availability of the directory | Transport availability | Data store availability | Lowest value |
| | Traceability for selecting source or destination | Transport traceability | Entry traceability | Lowest value |

### 10.4.2 **Comparing ALP effectiveness in scenario 1**

Scenario 1 is concerned with short term ALPs whereby layup activities need to be monitored. The following two sections will discuss the comparison of ALP effectiveness in reporting error information to the Line lead, and in the line lead trying to obtain information on the activities, respectively.

#### 10.4.2.1 Obtaining activity state and progress information

In the cases where the line lead needs to obtain activity state and progress information Figure 59 shows the ALPs available. Table 29 shows the combined effectiveness scoring for ALP_1.1 in this case and similar tables are available in Appendix F for ALP_1.2 and ALP_1.3. Table 30 shows a comparison between the combined effectiveness ratings. The data source is known through training and therefore the data locating mechanism is not part of the ALP.

**Table 29 – ALP_1.1 effectiveness scoring for obtaining activity state information**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~ 25 min | ~0.1 min | ~25 min |
| | Bandwidth | limitless | limitless | limitless |
| Accuracy | Degradation | | ~2%/min | ~2%/min |
| | Compression | ~50% | ~50% | ~25% |
| | Success rate | | | ~60% |
| | Correct destination | 100% | | 100% |
| | Data recentness | | 2/3 | 2/3 |
| | Size limit | ~33 bytes | ~33 bytes | ~33 bytes |
| Visibility | Availability | ~6% | ~14% | ~6% |
| | Traceability | 1/3 | 1/3 | 1/3 |

**Table 30 – Comparison of ALPs for obtaining activity state information**

| Characteristic | Property | ALP_1.1 | ALP_1.2 | ALP_1.3 |
|---|---|---|---|---|
| Responsiveness | Time | ~25 min | ~3 min 25 sec | ~2.5 sec |
| | Bandwidth | limitless | ~5 Mb/s | ~ 5 Mb/s |
| Accuracy | Degradation | ~2%/min | 0%/min | 0%/min |
| | Compression | ~25% | ~10% | ~10% |
| | Success rate | ~60% | 100% | ~100% |
| | Correct destination | 100% | 100% | ~100% |
| | Data recentness | 2/3 | 2/3 | 3/3 |
| | Size limit | ~33 bytes | ~33 bytes | 16+ MB |
| Visibility | availability | ~6% | ~14% | ~99% |
| | traceability | 1/3 | 1/3 | 3/3 |

#### 10.4.2.2 Reporting activity errors

Figure 60 shows the AL mechanisms available to a worker when the Line lead must be notified of an error during an activity. Table 31 shows the combined effectiveness of the AL mechanisms involved in ALP_2.1. Since the Line lead knows the location of the moulding shop and WhatsApp group, locating data sources does not form part of the ALPs. Similar tables are available for ALP_2.2 and ALP_2.3 in Appendix F. Table 32 compares the effectiveness of ALPs available for this case.

**Table 31 – ALP_2.1 effectiveness score reporting activity errors**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~ 25 min | ~8 s | ~25 min |
| | Bandwidth | ~7.52 b/s | ~7.52 b/s | ~7.52 b/s |
| Accuracy | Degradation | ~2%/min | ~2%/min | ~2%/min |
| | Compression | ~50% | ~50% | ~75% |
| | Success rate | | | ~99% |
| | Correct destination | ~99% | | ~99% |
| | Data recentness | | 2/3 | 2/3 |
| | Size limit | ~33 bytes | ~33 bytes | ~33 bytes |
| Visibility | availability | 14 % | 14 % | 14 % |
| | traceability | 1/3 | 1/3 | 1/3 |

**Table 32 – Comparison of ALPs for reporting activity errors**

| Characteristic | Property | ALP_2.1 | ALP_2.2 | ALP_2.3 |
|---|---|---|---|---|
| Responsiveness | Time | ~25 min | ~13 min | ~2.5 sec |
| | Bandwidth | ~7.52 b/s | 1 kb/s | ~ 5 Mb/s |
| Accuracy | Degradation | ~2%/min | 4%/min | 0%/min |
| | Compression | ~75% | 50 % | ~10% |
| | Success rate | ~99% | ~99% | ~99% |
| | Correct destination | ~99% | ~99% | ~99% |
| | Data recentness | 2/3 | 1/3 | 3/3 |
| | Size limit | ~33 bytes | 33 bytes | 16+ Mb |
| Visibility | availability | ~8 % | ~14% | ~99% |
| | traceability | 1/3 | 1/3 | 3/3 |

### 10.4.3 **Comparing ALP effectiveness in scenario 2**

The following sections present the comparison of the effectiveness of the ALPs observed in scenario 2. The effectiveness score table for the most effective ALP observed in normal operation is shown, followed by a comparison table for all observed ALPs.

10.4.3.1 Obtaining Standard Work data on activities

The only process that was available to obtain Standard Work information with normal operation was for a time study to be done by an industrial engineer with a stopwatch. In this case it required the full attention of the person for the entire day, limited by the amount of activities they can successfully monitor and the amount of data they can record by hand.

Since this information is not available until it is placed on the shared drive, the transport time will be set to 8 hours. Considering the time it takes to create the spreadsheet to store the data, the data storage time is estimated at around 3 hours. The size limit will be set at the total amount of data recordable by hand for 8 hours. It should also be noted that if the person needs to split attention between activities, this size limit will be split too. It will be assumed all needed data can be recorded without compression to the spreadsheet.

Table 33 shows the combined effectiveness metrics for the normal standard work obtaining ALP and Table 34 compares this with BASE equipped with a Standard Work plugin. Figure 61 illustrates these ALPs and their component mechanisms.

**Table 33 – ALP_ 3.1 effectiveness for obtaining Standard Work information**

| **Characteristic** | **Property** | **Transport** | **Store** | **Combination** |
|---|---|---|---|---|
| Responsiveness | Time | ~ 8 h | ~ 3 h | ~11 h |
| | Bandwidth | 7 bytes/s | ~7 bytes/s | ~7 bytes/s |
| Accuracy | Degradation | 0 MB/s | 0 MB/s | 0 MB/s |
| | Compression | 0% | 0% | 0% |
| | Locating Success rate | | | ~99% |
| | Correct destination | ~99% | | ~99% |
| | Data recentness | | 1/3 | 1/3 |
| | Size limit | Limitless | ~10 MB | ~10 MB |
| Visibility | Function availability | Once a year | ~99% | Once a year |
| | Function traceability | 2/3 | 2/3 | 2/3 |

**Table 34 – ALP comparison for obtaining Standard Work information**

| Characteristic | Property | ALP_3.1 | ALP_3.2 |
|---|---|---|---|
| Responsiveness | Time | ~11 h | ~0.5 s |
| | Bandwidth | ~7 bytes/s | ~5 MB/s |
| Accuracy | Degradation | 0 MB/s | 0 MB/s |
| | Compression loss | 0% | ~10 % |
| | Locating Success rate | ~99% | ~99% |
| | Correct destination | ~99% | ~99% |
| | Data recentness | 1/3 | 3/3 |
| | Size limit | 0.2 MB | 16+ MB |
| Visibility | Availability | Once a year | ~99% |
| | Traceability | 2/3 | 3/3 |

10.4.3.2 Tooling maintenance

For tooling maintenance and information, the responsibility was placed on workers and the line lead to manage the information. The available ALPs to obtain the information is illustrated in Figure 62 and the effectiveness of ALP_4.1 is shown in Table 35, with a comparison of all ALPs in Table 36.

**Table 35 – ALP_4.1 effectiveness tooling and maintenance information**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~25 min | ~ 4 s | ~25 min |
| | Bandwidth | limitless | ~8 bytes/s | ~8 bytes/s |
| Accuracy | Degradation | 0 %/min | ~ 2%/min | ~2%/min |
| | Compression loss | 0% | ~ 50% | ~50% |
| | Locating Success rate | | | ~60% |
| | Correct destination | ~99% | | ~99% |
| | Data recentness | | 1/3 | 1/3 |
| | Size limit | limitless | 33 bytes | 33 bytes |
| Visibility | Function availability | ~8 % | ~99 % | 8 % |
| | Function traceability | 1/3 | 1/3 | 1/3 |

**Table 36 – Comparing ALPs for obtaining tooling and maintenance information**

| Characteristic | Property | ALP_4.1 | ALP_4.2 | ALP_4.3 |
|---|---|---|---|---|
| Responsiveness | Time | ~25 min | ~13 min | ~2.5 sec |
| | Bandwidth | ~8 bytes/s | ~7.52 bytes/s | ~ 5 Mb/s |
| Accuracy | Degradation | ~2%/min | ~4%/min | 0%/min |
| | Compression | ~50% | ~50 % | 10% |
| | Locating Success rate | ~60% | ~99% | 99% |
| | Correct destination | ~99% | ~99 % | ~99% |
| | Data recentness | 1/3 | 1/3 | 3/3 |
| | Size limit | 33 bytes | ~5 MB | 16+ Mb |
| Visibility | availability | ~8 % | ~14% | ~99% |
| | traceability | 1/3 | 1/3 | 3/3 |

10.4.3.3 Worker feedback and quality improvement

The ALP mechanisms involved with worker feedback and quality improvement is illustrated in Figure 63. Table 37 shows the combined effectiveness rating for ALP_5.1 and Table 38 compares all observed ALPs for this business process. A worker wanting feedback on a part requires them to actively search for the right person or report. The success rate for this location mechanism will be placed at 10% (based on one worker, out of the five that are reporting, being able to perform this activity about twice a week). For a person to give feedback, the worker's station is known, and the locating function will not play a role.

**Table 37 – ALP_5.1 effectiveness for obtaining worker feedback information**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~ 25 min | ~ 0.5 min | ~25 min |
| | Bandwidth | ~8 bytes/s | > 10 Mb/s | ~8 bytes/s |
| Accuracy | Degradation | 0 %/min | ~2 %/min | ~2 %/min |
| | Compression | ~50 % | ~50 % | ~50 % |
| | Success rate | | | ~10 % |
| | Correct destination | ~99 % | | ~99 % |
| | Data recentness | | 1/3 | 1/3 |
| | Size limit | 33 bytes | > 10 MB/s | 33 bytes |
| Visibility | Function availability | ~14% | ~14% | ~14% |
| | Function traceability | 1/3 | 1/3 | 1/3 |

**Table 38 – Comparison of ALPs for worker feedback and quality improvement**

| Characteristic | Property | ALP_5.1 | ALP_5.2 | ALP_5.3 |
|---|---|---|---|---|
| Responsiveness | Time | ~25 min | ~13 min | ~2.5 s |
| | Bandwidth | ~8 bytes/s | ~ 8 bytes/s | 16+ MB/s |
| Accuracy | Degradation | ~2 %/min | ~2 %/min | 0 MB/s |
| | Compression | ~50 % | ~25 % | ~10 % |
| | Success rate | ~10 % | ~10 % | ~99 % |
| | Correct destination | ~99 % | ~99 % | ~99 % |
| | Data recentness | 1/3 | 1/3 | 3/3 |
| | Size limit | ~33 bytes | ~33 bytes | 16+ MB |
| Visibility | availability | ~6% | ~14% | ~99 % |
| | traceability | 1/3 | 1/3 | 3/3 |

### 10.4.4 **Discussion**

The ALPs observed at the industry partner were a complex mixture of human-to-human communication, notice boards and visual observations. Many of these ALPs required a worker or Line lead to walk to a source or sink of commonly used and critical information. From the comparison tables given, a notable reduction in time to obtain or deliver the same information can be seen. The benefit of connecting workers to the digital infrastructure removed barriers of time and space for communication, and the structuring of data flow was automated and governed by the BASE architecture principles.

The traceability of resources and activities were improved when they formed part of the BASE holarchy since this information is digitised at source, stored in the same place for all holons, and the entries were all formatted to the same structure. This strong improvement in visibility and responsiveness is attributable to the use of digital communication and storage mechanisms governed by the generalised and federating nature of the BASE holarchy.

The BASE Service Provision Contract allowed the automation of ALPs for orchestrating resources and activities. If the same amount of data was recorded manually, it would add an unfeasible amount of repetitive administrative tasks to workers who would not only suffer in productivity but also reduction in attention with a likelihood of increase in mistakes. The BASE implementation showed the ability to digitize this execution data at the source, allowing automation of all further ALPs on this data.

The analysis also showed that many ALPs that end with a human memory storage mechanism suffer in the long term due to memory degradation. This is observed as workers forgetting feedback, training instructions, or management forgetting about reported problems or suggestions.

## 10.5 Benefits of the BASE Architecture

Following the comparison of ALP effectiveness for the scenarios in the case study, Section 10.5 will discuss the observed benefits of the BASE architecture. Included in the discussion is its effects on traceability, process clarity, cost per decision among other quantitative benefits. Also discussed in this chapter are more qualitative benefits of BASE during development, scaling, and maintenance compared to traditional implementations of such software systems.

### 10.5.1 **Lowering cost per decision**

An aspect not discussed in the analysis is that of an ALPs cost. Time was one of the properties measured as part of responsiveness, but it also indicates person-hours lost between decision making points. When the ALPs are handled by the BASE holarchy, the cost of obtaining this data becomes insignificant, lowering the cost per decision for humans and allowing new opportunities in decisions and process optimization. As stated in Section 2.4, a human resource is a decision resource; therefore, lowering the cost of decisions means enhancing the value of humans.

As an example, considerer the walk-to-see AL mechanism, which seems attractive due to its high accuracy. However, it was observed that many decisions revolve around fast access to simple data types (e.g. Boolean or string) for which the accuracy of this mechanism is excessive. The decision to use or not use this mechanism was not required prior to the implementation of the BASE architecture.

Another example is the cost of obtaining Standard Work data. The normal manual method of recording this data meant that it could only be afforded once a year. With the automatic capturing and provision of this data through the BASE administration shells, opportunities were created to apply statistical analysis on this data from the large sample sets that could be collected. The industry partner expressed that they wish to use this data to determine when process maturity is reached, based on its changing standard deviation attribute.

### 10.5.2 **Improving process clarity**

The business processes discussed in the case study are linear, well-understood events. There are events beyond the processes discussed that range from engineering changes and customer feedback, to supply chain optimization, for which the ALPs involved are more complex. With the BASE administration shell applied to assets, high-fidelity process clarity is achieved in real time, as well as historically. Factory scheduling and planning benefits in turn, and the industry partner mentioned that they see a possibility to draw correlations between cause and effect in the factory previously impossible due to the temporal and spatial gap between events, resources and the analysis.

### 10.5.3 **Unifying and integrating resources by federating ALPs**

The implementation of the BASE architecture demonstrated that it can be applied to tools, activities and humans. When communication and orchestration occur through BASE administration shells, the AL involved is federated. This removes physical and communicative barriers, allowing people to communicate with tools and activities the same way that tools and activities can now communicate with people.

Without the BASE administration shell, a worker needs to take on administrative responsibilities on behalf of non-influential entities, such as moulds or ply packs. When these resources are equipped with administration shells, they are given a voice and the channels to speak to humans and other influential entities. This can automate much of the administrative load on workers, creating a better work experience and removing human error.

### 10.5.4 Reflecting reality versus recording reality

Much of the benefit the BASE administration shell offers come from the digital nature of its storage and communication channels. The question is then why the BASE holarchy is better than a traditional database and centralised server.

The simple answer is that reflecting reality is better than emulating it (Valckenaers & Van Brussel, 2016). The structure of reality in manufacturing is that of entities organising and communicating, based on activities that need to be performed. An abstraction of this data into a database means the following:

- Restructuring of the organisation or bringing in new types of resources or activities means manual restructuring of the database, data collection and data flows between systems.
- Business structures and decisions have limits imposed by the rigidity of a centralised database and decision-making system.
- Communication or information required by one entity of another requires full understanding of how to search the database.
- Unique, localised decision-making capabilities cannot be applied directly to an entity but must be incorporated into the monolithic centralised program.
- Communication channels all converge to a single point, traveling through many middlemen instead of occurring directly between entities involved.
- No offline capabilities when the central server goes down. [6]

When looking at literature that discusses control and organisational structures, similar views are expressed. Simon (1997) discusses qualitative properties that make an organisation's administration more effective by:

- Specialisation – A collective term summarising the properties of reconfigurability, modularisation, and separation of concerns.
- Unity of command – Many smaller modular components require good organisation of their specialisations, and the ability to work in unity for the goal at hand.
- Reducing span of control – This can be seen as specialisation in authority which reduces micromanagement and allows a reduction in the complexity of the world of interest on authoritive decision makers.

[6] This ultimately caused the downfall of the Federation when Anakin destroyed their droid control ship.

These three principles are core to distributed control. These principles are also directly related to the characteristics of a holonic control implementation as described by Kruger and Basson (2019). This distribution minimises the organisational levels to go through before action can be taken. Smaller intelligent modules working together supports the horizontal specialisation in an organisation, but also allows dynamic specialisation in the vertical aspect of organisational structure.

The characteristics described by Kruger and Basson (2019) not only relate to the effectiveness of the implementation's function in administrative aspects, but plays a large role in how well the architecture facilitates development and deployment of the system – as discussed in section 10.5.6.

### 10.5.5 **BASE holarchy scalability**

The case study started by applying a BASE administration shell to one worker, and incrementally scaling the holarchy from there. Two workers were added the next day, and by the first week the moulds were also incorporated into the holarchy. Plugins were also incrementally added since activity data is fully encapsulated by the 3SAL structure and temporally separated between the three stages.

Figure 66 illustrates the structure of BASE holarchy if it was expanded to encapsulate more activities and resources. The moulding holarchy follows the same principles as the higher level holarchies it serves, and this recursive pattern of expansion can theoretically be continued on a per BASE administration shell level.

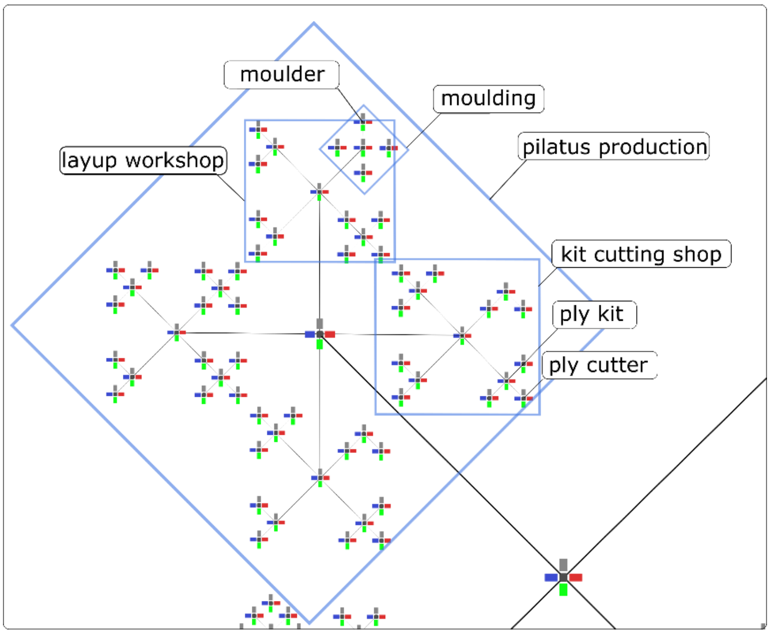



**Figure 66 – An illustration a BASE holarchy if expansion was continued.**

### 10.5.6 **Supporting development and implementation**

The metrics discussed in Section 10.1 to 10.4 described the system in a working state. There is also a benefit to using BASE architecture in planning and structuring the research, development and implementation of both administration shells and plugins.

#### 10.5.6.1 Concurrent development and maintenance

The architecture shows high modularity. It offered a clear separation of functional components where core components offered single source of truth on all decision-making data. Since each core data storage component is maintained by only the anti-clockwise adjacent plugins, a firewall was created between corners against potentially negative effects of upgrading, changing, or adding new plugins.

The more specialised and independent components of a system is, the more of them can concurrently be developed and maintained. The BASE architecture not only allows administration shells to be independently developed, but also allows the individual business decisions of those shells to separate into the four plugin components.

Monolithic programs with many unique interacting components very quickly limit the amount of developers that can work concurrently on the system due to the complexity that arises. If by adding twice as many developers the development time is reduced by half, the development has a 1:1 scaling ratio. It is believed that the BASE architecture implementation comes very close to this ideal if all developers follow the same architectural guidelines.

The architecture also showed high integrability and convertibility. Since plugins communicate through a standardised activity structure and holons communicate through dynamic establishment of service contracts, the architecture is highly integrable and supports online upgrades and holon additions. The BASE architecture is structured in a manner that does not require a complete implementation running for the benefits of its components to start showing.

#### 10.5.6.2 Verification

To verify that the system works, is reliable and fulfils its goals, is often not feasible to do on a full-scale implementation. When components are specialised, they require specialised tests. Generalising the communication and orchestration between components through the BASE architecture allows for faster and more reliable testing.

However, the architecture is not the only factor affecting verification – as Kruger and Basson (2019) stated, the choice in programming language plays a key role in how verifiable an implementation is. Based on their recommendation, and as was experienced by the author, the use of the Erlang programming language was a decision that greatly facilitated verification.

10.5.6.3 Reusability

Re-usable code improves development time and reliability, since less individual components need to be checked and verified or maintained. Re-usable code also implies better modularity and specialisation. Separating business decisions into the four plugin components and federating the communication between them through 3SAL facilitates the re-usability of code between BASE administration shells.

### 10.5.7 **Federating ALPs creates an integration platform**

Organisations often require specialists to consult them on developing custom integration tools to federate different software packages, databases, and resource management tools. The problem of integration can be considered a problem in effective ALP's between these discission making tools. The same problem applies when trying to integrate new tooling, interfaces, or other hardware from different vendors with proprietary software. Often a custom linking software will be commissioned to combine these items for one specific use case, and require a new piece of software be developed for a different use case. When each resource has a unified ALP handling mechanism, these problems effectively disappear.

10.5.7.1 Integrating resource management systems

With recent developments in I4.0 enabling technologies, many companies have developed data analysis software packages to optimise processes and business decisions. SAP is a well-known tool that is implemented in companies for inventory management and cost calculations and relies on accurate and up-to-date data for its models to be effective. The BASE architecture can allow for integration with, and the support of, such tools. In this case, an AP can be developed to automatically deliver information to the SAP system, adding value to the SAP models and calculations with increased fidelity an accuracy.

The BASE architecture's capabilities were demonstrated to Srihari Murthy, CEO of Stigmergy solutions – a company that specialises in the digitisation and implementation of lean manufacturing principles in the composites industry. Stigmergy utilises the Compass analysis software developed by Visual8, which relies on high fidelity and real-time shop floor data for process planning and optimisation. Murthy mentioned that the implementation of the BASE architecture would allow them to develop models with higher resolution and better optimisation (Murthy, 2020).

The BASE architecture also supports integration with existing Scheduling tools. For example, the Schedule of the HRH could be easily integrated with an application such as Google Calendar, using the Google APIs, or draw data from a spreadsheet.

10.5.7.2 Integrating interfacing technologies, tools, and hardware

Recent developments in advanced human interfacing technology shows highly configurable and capable devices for augmented reality, haptic feedback and voice commands. The high configurability and data bandwidth of these devices require an equally high level of orchestration and dynamic management. The BASE architecture has demonstrated the ability to dynamically orchestrate human interfaces based on their capabilities and the situational requirements (Sparrow *et al.*, 2020).

Further discussion with Murthy on their development of the company *Factri.ai* which specialises in introducing I4.0 and IoT technologies to SMEs in India revealed the need for their different software packages and modular hardware to be integrated on a common platform. Due to the diversity of companies they work with, each company requires a custom assembly of these products, and interest was shown in the possibility of giving each of their products and workers a BASE shell to improve their development and experimentation cycle times.

10.5.7.3 Incremental deployment

When a system is developed with a high concurrent development score, parts of the system can come online and potentially already deliver some of the benefits before the full system is deployed. If the system can be deployed in small steps, replacing or augmenting functionality without causing breaking disturbances, it has a high chance of successful adoption.

Incremental deployment also means that organisations do not have to invest a large amount of resources into something they still need to try out. It will allow a larger number of organisations to experience the potential benefits, and budget further deployments according to their means.

10.5.7.4 Time to integration

A new worker takes time to integrate with his environment, which can roughly be seen as the time it takes for the employee to show a net positive addition to the company’s value stream. Training, establishing good working practices and becoming familiar with the organisation’s structure and resources all take time before the employee can effectively contribute to the organisation. If a worker needs to work with a new machine, this training can be seen as internalising the required AL of translating and transporting data to the machine in a way that it interprets and acts on it correctly. This creates difficulty when workers need to be replaced or moved.

The effect that a BASE administration shell has in this regard will be a mix of qualitative and quantitative measures in the accuracy of their work, their comfortability in being assigned tasks and reduction of re-work.

10.5.7.5 Integration with existing systems

The cost of overhauling a manufacturing system is critical to the success of the new system being implemented, regardless of the improvements it can offer. Many companies cannot afford new equipment or infrastructure, and the implemented holonic control system should ideally be compatible with existing networks, machines and infrastructure, or provide a means to raise existing resources to a level compatible with the holonic control system as with Administration shells.

### 10.5.8 **Benefit to human factors**

A major benefit of the HRH-AS will come from the resulting human experience. The following metrics are qualitative and cannot, or should not, necessarily be quantitatively determined as they represent the emotional and personal gain the administration shell brings to the worker using it, which could show benefit regardless of the quantitative measures given in Section 10.

10.5.8.1 Value-adding time gained

When a worker needs to file reports, walk to find a foreman, struggle to find data or perform other non-value adding activities related to logistics and not decision making, improvements in the AL will see an improvement in the ratio of value-adding work done versus non value-adding work, which can be measured as a change in percentage.

10.5.8.2 Worker empowerment

Workers at the bottom of an organisational structure may have important information or suggestions on products or production, but the available communication channels can often make them feel that their opinion is not valued. Furthermore, workers often report this information to a foreman verbally, who has multiple other obligations and needs to prioritise them, during which time the information is either forgotten or does not make it to the correct destination in time. Workers with ideas or suggestions are often stifled by the bureaucracy of message channels to get ideas to relevant decision makers in the organisation. If they are able to express their opinions and know that their message won’t get lost in the system, they may feel more empowered and willing to contribute. Worker empowerment is a qualitative metric, and its value lies in the opinion of the workers.

10.5.8.3 Worker enjoyment

Improved worker enjoyment drastically improves the quality of data they collect and their time to adopt a new system. The metrics presented in Section 10 may quantitatively show how certain aspects of administrative logistics are improved and facilitate with worker integration, but the complexities of human interaction with systems cannot empirically tell whether the worker perceives the system to help or hinder them – only the worker can. Therefore, a worker equipped with a BASE administration shell can give feedback and insight that will provide invaluable insight into the quality the administration shell adds to their work process.

### 10.5.9 **Effective AL for Big Data**

Yu (2015) explained three critical steps in Big Data analytics comes from collecting, refining, and delivering data. These three steps require ALPs and due to the different mechanisms used can cause extensive problems for Big Data analytics:

> *"Most data sets in our age of 'Big Data' are utterly inadequate for advanced analytics, let alone end-users. That is the primary reason why most analysts spend 50 to 80 percent of their valuable time fixing and preparing unstructured, uncategorized, and unrefined raw data sets. The situation is not much different from trying to run a brand new sports car with unrefined oil… Serious data scientists need to make data refinement their first priority, and break down the data work into three steps: Collection, Refinement, Delivery."* (Yu, 2015)

The BASE architecture was shown to format and collect data on resources and activities in a single standard 3SAL data structure with time-stamps and available data on each distinct life-cycle stage. The analysis plugins define a clear location for these analysis tools and where they would obtain their data from.

## 10.6 Shortcomings of the BASE architecture

The following subsections discuss some shortcomings in the architecture that were identified while developing the case study implementations. The identified shortcomings are related to the BASE architecture and aspects regarding its implementation and indicate where further research and development is required.

### 10.6.1 Complexity of the Execution component

Compared to the data repository functions of the Biography, Attributes, and Schedule, the Execution component has a disproportionately more complex responsibility. This is not an inherent problem with the BASE architecture, but is rather caused by the high-density information flow that occurs at the interface between the physical and virtual worlds at the origo. The issue is exacerbated by the need for near real-time communication as activities approach to the origo. It is recommended that further research be conducted on the internal architecture, as well as the implementation, of the Execution component.

### 10.6.2 Interacting with resources without a BASE administration shell

The benefits of implementing the BASE architecture as a digital administration shell can only be fully realised when other resources in the environment are also equipped with administration shells. If a worker equipped with a BASE administration shell operates within an environment where there are no other digitally enabled resources, the value of the implementation will be limited to analysis and planning only on information in its core components. As such, value-addition through the BASE architecture relies on development of external systems.

### 10.6.3 No guidance to hardware or infrastructure needs

In its attempt to stay agnostic of hardware and vendor capabilities, the BASE architecture focused on the functional aspects of the architecture and presents some difficulties when hardware or infrastructure needs to be chosen for components. As an example, there is no clear line for where local or cloud storage needs to be used and, in this sense, the core components may be oversimplified and may need further internal refinement to generalise hardware or infrastructure requirements.

### 10.6.4 Requirement for business process knowledge

When developing and growing a holarchy to encompass business processes, it takes time to understand what activities need to orchestrate which resources. Often there is no clear mapping of processes on a shop floor and without this standardisation it may be difficult to create BASE holons that will accurately reflect the shop floor.

In some cases efficiency was obtained with ad-hoc methods thought up by workers and when implementing a BASE holarchy, a globally optimal solution in a process may need to be reduced to be more locally optimal in order for a resource to take part in the holarchy. To make such decisions requires in depth understanding of the existing business processes involved.

### 10.6.5 Architecture maturity

The BASE architecture was created from advanced manufacturing concepts such as holonic design principles, distributed control, and other I4.0 concepts. Without fully understanding the direction the architecture is intended to take, and how it should be applied, it can easily lead to poor performing systems, or very complex, unmaintainable structures, when traditional software design principles and engineering is applied.

# 11 Conclusion

The BASE architecture presented in this dissertation aims to address the challenge of human integration in emerging I4.0 environments. It was proposed that the processes referred to as AL are the largest contributing factor that hinders effective human integration. The BASE architecture provides a structured method of implementing a RAMI 4.0 administration shell to elevate the human to resource holon status, and facilitates the handling of AL needed to integrate the humans with digital systems around them. The architecture provides digital augmentations, which work alongside the human to support scheduling, biographing, execution, and modelling functions, while maintaining a communication interface with digital systems.

## 11.1 Dissertation summary

In this dissertation, a literature study was conducted (and summarized in Section 1) after it was determined that there is a need for research to better integrate workers with an I4.0 manufacturing environment. The literature review explored modern manufacturing methodologies and technology as well as concepts of organisational structure and administration to better understand the challenges this would present.

The dissertation then explored the integration issues from the knowledge of the literature study to uncover the underlying cause of integration issues. The dissertation identified key concepts in an organisations structure and administration and broke these into two parts, which were called Administrative Cognition and AL. Since humans are regarded as excellent decision makers, the blame on unsuccessful integration was determined to be related to AL.

It was then determined that a RAMI4.0 digital administration shell concept would allow the human worker to be elevated to HCPS status and facilitate in the handling of AL involved with integration into I4.0 environments.

With the added benefit of digital data processing and communication, a Holonic Manufacturing System approach was taken to structure the required functionality of the administration shell. This led to the concept of the Human Resource Holon Administration shell discussed in Section 4.

The BASE architecture was developed to guide the creation of a HRH-AS as discussed in Section 6. During the development of the BASE architecture, fundamental questions on the nature of activities and their executors needed to be answered, leading to the development of the 3SAL activity structure discussed in Section 5.

Section 7 discussed the implementation of a BASE administration shell core. The core components of BASE are designed to be generic to any application, requiring only the plugin components that contain the business decisions to modify a BASE instance for a specific application.

Further refinement of the BASE Execution component was required to fully realise the human interfacing requirements of the HRH-AS. This resulted in the development of the semiotic and observation interface holon concepts. These interface holons allowed for flexible and robust interfacing with the human following holonic principles and was discussed in Section 8.

Section 9 discussed the development and implementation of two case studies to fully evaluate the performance of a BASE architecture implementation. A case study partner was selected and with their help the case studies were developed to bring benefit to their existing business processes and test how BASE facilitated this process.

Section 10 evaluates the performance of the BASE architecture in its fulfilment of the three HRH-AS responsibilities, the value-addition to the industry partner and its ability to elevate the human to Resource Holon status. Some qualitative benefits of using the BASE architecture as a guide to develop an HRH-AS are also discussed in Section 10.

## 11.2 Evaluation summary

The evaluation was aimed at determining how well the BASE implementation fulfilled its three HRH-AS responsibilities. The first case study showed the architecture capable of orchestrating interface services and allowed for the intelligence surrounding this orchestration to be changed and upgraded with the EPs.

Further evaluation required that the effect that the BASE architecture implementation had on AL be measured to determine if it could facilitate human integration in this aspect. A set of metrics were formulated to measure the effectiveness of four identified Administrative Logistic Functions and their execution through specific mechanisms. The Administrative Logistic Processes observed in the second case study were broken down into the ALFs they consist of and could subsequently be evaluated in terms of their effectiveness. The AL effectiveness for each scenario in case study two was also evaluated when assets were equipped with BASE administration shells and tables comparing the ALP effectiveness for all scenarios were created.

The evaluation concluded that BASE had a dramatic effect on the responsiveness and accuracy of the ALPs observed, and could lift some of the AL responsibilities of tooling maintenance. The implementation also showed the ability to automate the capturing and analysis of Standard Work information, allowing for new business decisions based on statistical analysis of this data not possible before.

## 11.3 Recommendations for further research

The following sections identify and discuss aspects of this research that need to be expanded on or further studied.

### 11.3.1 Architectural refinements to ensure unanimous development

Due to the generalisation made to create the BASE architecture, research needs to be conducted into the refinement of the internal structure of the core components. Specifically, to specify what parts of activity execution are common among all resource types, as well as what refinements to the Attributes component will be applicable to all applications.

### 11.3.2 Mapping BASE to ARTI

During the research it was found that the BASE architecture very closely maps to the ARTI reference architecture. Exploring the connections and parallels will allow research based on holonic architectures, and specifically the BASE architecture, to contribute in further developments and implementations.

### 11.3.3 Further development of interface holons

The concepts of holons that provide observation and semiotic services need to be further developed to ensure their effective integration with HRH-ASs. Concepts from the study of semiotics, the psychology of information delivery through human senses and how to utilise multiple interface holons to maximise effective communication needs to be included in this research. Along with this research, a Service Description Language (SDL) needs to be developed for interface holons. The SDL will be used to identify and differentiate interfaces and their capabilities and will be vital to the Execution component when making decisions on which services to use to obtain or deliver information. The author has started to explore this problem and found the Resource Description Framework and OWL ontologies to be promising.

To ensure the effective decision-making of the plugin components, as well as the internal workings of the Execution component, research into creating models of the human worker specifically for the manufacturing environment needs to be conducted. The relationship between the physical, mental, and biological models with relation to the model accuracy, processing constraints and priority needs to be explored. A common ontology for communicating these human models with other CPSs in the manufacturing environment may be needed.

### 11.3.4 Relating hardware requirements to BASE components and implementations

The relationship between hardware requirements and the nature of data components of BASE over time needs to be explored. The Attributes component can potentially hold large amounts of data, and since BASE internalises this data for each of potentially thousands of holons in a holarchy, a method to handle this kind of data storage requirement needs to be explored.

The Communications Manager was not developed to consider critical aspects of security and other problems related to connectivity to the internet. The Communications Manager is tasked with an undetermined load of concurrent conversations from internal plugins and external connections, and research on the types of technology available to handle these requirements is needed.

# 12 References

Aaron, P. & Lane, W. 2017b. *Industry 4.0 and the digital twin Manufacturing meets its match*. Deloitte Insights: Deloitte. [Online], Available: https://www2.deloitte.com/us/en/insights/focus/industry-4-0/digital-twin-technology-smart-factory.html [2018, June 21].

Ashimura, K. & Dahl, D. n.d. *W3C Multimodal Interaction Working Group*. [Online], Available: https://www.w3.org/2002/mmi/ [2019, February 16].

Åstrand, P.-Olof. & Rodahl, K. 1986. *Textbook of work physiology : physiological bases of exercise*. McGraw Hill.

Baheti, R. & Gill, H. 2011. Cyber-Physical Systems. The Impact of Control Technology. *The impact of control technology*. 12(1):161–166.

Bains, Paul. 2006. *The primacy of semiosis: An ontology of relations*. Toronto: University of Toronto Press.

Barricelli, B.R., Casiraghi, E., Gliozzo, J., Petrini, A. & Valtolina, S. 2020. Human Digital Twin for Fitness Management. *IEEE Access*. 8:26637–26664.

Bertram, P., Birtel, M., Quint, F. & Ruskowski, M. 2018. Intelligent Manual Working Station through Assistive Systems. *IFAC-PapersOnLine*. 51(11):170–175.

Brussel, H.V., Wyns, J., Valckenaers, P., Bongaerts, L. & Peeters, P. 1998. Reference Architecture for Holonic Manufacturing Systems : PROSA Reference Architecture for Holonic Manufacturing Systems : PROSA. 37(3):255–276.

Burns, M., Manganelli, J., Wollman, D., Laurids Boring, R., Gilbert, S., Griffor, E., Lee, Y.-C., Nathan-Roberts, D., et al. 2018. Elaborating the Human Aspect of the NIST Framework for Cyber-Physical Systems. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*. 62(1):450–454.

Calzavara, M., Persona, A., Sgarbossa, F. & Visentin, V. 2018. A model for rest allowance estimation to improve tasks assignment to operators. *International Journal of Production Research*. 0(0):1–15.

Cardin, O., Derigent, W. and Trentesaux, D., 2018. Evolution of holonic control architectures towards Industry 4.0: A short overview. IFAC-PapersOnLine, 51(11), pp.1243-1248.

Chaitin, G. 2006. The Limits of Reason. *SCIENTIFIC AMERICAN*. 294(3):74–81.

Doshi, A., Smith, R.T., Thomas, B.H. & Bouras, C. 2017. Use of projector based augmented reality to improve manual spot-welding precision and accuracy for automotive manufacturing. *International Journal of Advanced Manufacturing Technology*. 89(5–8):1279–1293.

Elkosantini, S. 2015. Toward a new generic behavior model for human centered system simulation. *Simulation Modelling Practice and Theory*. 52:108–122.

Flemisch, F., Heesen, M., Hesse, T., Kelsch, J., Schieben, A. & Beller, J. 2012. Towards a dynamic balance between humans and automation: authority, ability, responsibility and control in shared and cooperative control situations. *Cognition, Technology & Work*. 14(1):3–18.

Geissbauer, R., Vedso, J. & Schrauf, S. 2016. Industry 4.0: Building the digital enterprise. *PwC*. 1–36.

Grenoble, L.A. 1998. *Deixis and information packaging in Russian discourse*. J. Benjamins.

Grieves, M. & Vickers, J. 2016. DigitalTwin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems (Excerpt). *Origins of the Digital Twin Concept*. (August):1–8.

Haines, R.F. & Chuang, S. 1992. *The effects of video compression on acceptability of images for monitoring life sciences experiments*. [Online], Available: https://ntrs.nasa.gov/citations/19920024689 [2020, September 21].

Hawkridge, G., Basson, A. & Kruger, K. 2019. Comparison of Erlang/OTP and JADE implementations for standby redundancy in a holonic controller. *International Journal of Computer Integrated Manufacturing*. 32:1–24.

Hedge, A. 2013. *Human information processing*. (University Notes). Cornell University. [Online], Available: http://ergo.human.cornell.edu/studentdownloads/DEA3250pdfs/HIP.pdf [2018, October 18].

International Institute of Business Analysis. 2006. A Guide to the Business Analysis Body of Knowledge. (1.6):329.

Ji, Z., Yanhong, Z., Baicun, W. & Jiyuan, Z. 2019. Human – Cyber – Physical Systems ( HCPSs ) in the Context of New-Generation Intelligent Manufacturing. 5:624–636.

Kritzinger, W., Karner, M., Traar, G., Henjes, J. & Sihn, W. 2018. Digital Twin in manufacturing: A categorical literature review and classification. *IFAC-PapersOnLine*. 51(11):1016–1022.

Kroemer, K.H.E., Kroemer, H.J. & Kroemer-Elbert, K.E. 2010. *Engineering Physiology*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Kruger, K. & Basson, A. 2015. An Erlang - based Resource Holon for a Holonic Manufacturing Cell. *Service Orientation in Holonic and Multi-agent Manufacturing*.

Kruger, K. & Basson, A.H. 2019a. Evaluation of JADE multi-agent system and Erlang holonic control implementations for a manufacturing cell. *International Journal of Computer Integrated Manufacturing*. 32(3):225–240.

Kruger, K. & Basson, A.H. 2019b. Evaluation criteria for holonic control implementations in manufacturing systems. *International Journal of Computer Integrated Manufacturing*. 32(2):148–158.

Lasi, H., Fettke, P., Kemper, H.G., Feld, T. & Hoffmann, M. 2014. Industry 4.0. *Business and Information Systems Engineering*. 6(4):239–242.

Lee, F. 2017. Ambient Intelligence - The Ultimate IoT Use Cases. [Online], Available: https://www.iotforall.com/ambient-intelligence-ami-iot-use-cases/ [2020, November 09].

Lee, J., Bagheri, B. & Kao, H.-A. 2015. A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems.

Leitão, P. & Restivo, F. 2006. ADACOR: A holonic architecture for agile and adaptive manufacturing control. *Computers in Industry*. 57(2):121–130.

Loch, F., Ziegler, U. & Vogel-heuser, B. 2018. Integrating Haptic Interaction into a Virtual Training System for Manual Procedures in Industrial Environments. *Proceedings of the 16th IFAC Symposium on Information Control Problems in Manufacturing*.

Lopes, I., Oliveira, A. & Costa, C.J. 2015. Tools for Online Collaboration: Do they contribute to Improve Teamwork? *Mediterranean Journal of Social Sciences*. (December, 10).

Loveday, S. 2018. *BMW Comments On Tesla Model 3 Production Woes*. [Online], Available: https://insideevs.com/bmw-comments-on-tesla-model-3-production-woes/ [2018, July 23].

McFarlane, D.C. & Bussmann, S. 2003. Holonic Manufacturing Control: Rationales, Developments and Open Issues. *Agent-Based Manufacturing*. 303–326.

Mitkin. 2018. *Erlang DRAKON plugin*. [Online], Available: http://drakon-editor.sourceforge.net/drakon-erlang/intro.html.

Monostori L, Kadar T, Kondoh S, Kumara S, Reinhart G, Sauer O, Schuh G, Sihn W, U.K. 2016. CIRP Annals - Manufacturing Technology Cyber-physical systems in manufacturing. 65:621–641.

Murthy, S. 2018. Personal interview. 20 December, Dubai. *"Interview on lean manufacturing and Industry 4.0 in the composites and SME environment"*

Murthy, S. 2020. Personal interview. 15 June, Cape Town. *"A discussion on the possibilities of the BASE architecture for SMEs and data capturing"*

Norvig, P. 2012. *English Letter Frequency Counts: Mayzner Revisited or ETAOIN SRHLDCU*. [Online], Available: http://norvig.com/mayzner.html [2020, September 21].

Nylund, H., Salminen, K. & Andersson, P. 2008. Digital Virtual Holons — An Approach to Digital Manufacturing Systems. in *Manufacturing Systems and Technologies for the New Frontier* London: Springer London. 103–106.

Pacaux-Lemoine, M.P., Trentesaux, D., Zambrano Rey, G. & Millot, P. 2017. Designing intelligent manufacturing systems through Human-Machine Cooperation principles: A human-centered approach. *Computers and Industrial Engineering*. 111:581–595.

Peden, R.G., Mercer, R. & Tatham, A.J. 2016. The use of head-mounted display eyeglasses for teaching surgical skills: A prospective randomised study. *International Journal of Surgery*.

Peruzzini, M., Grandi, F. & Pellicciari, M. 2019. Exploring the potential of Operator 4.0 interface and monitoring. *Computers and Industrial Engineering*. Vol 139:105600.

Posada, J., Toro, C., Barandiaran, I., Oyarzun, D., Stricker, D., De Amicis, R., Pinto, E.B., Eisert, P., et al. 2015. Visual Computing as a Key Enabling Technology for Industrie 4.0 and Industrial Internet. *IEEE Computer Graphics and Applications*. 35(2):26–40.

Quint, F. & Loch, F. 2015. Using smart glasses to document maintenance processes. *ensch und Computer 2015 Workshopband*. (September):203–208.

Redelinghuys, A., Basson, A. and Kruger, K., 2018, June. A six-layer digital twin architecture for a manufacturing cell. In International Workshop on Service Orientation in Holonic and Multi-Agent Manufacturing (pp. 412-423). Springer, Cham.

Rey, G.Z., Carvalho, M. & Trentesaux, D. 2013. Cooperation models between humans and artificial self-organizing systems: Motivations, issues and perspectives. *Proceedings - 2013 6th International Symposium on Resilient Control Systems, ISRCS 2013*. 156–161.

Riva, G., Vatalaro, F. & Davide, F. 2005. *Ambient Intelligence: The Evolution of Technology, Communication and Cognition Towards the Future of Human-computer Interaction*. US: IOS Press.

Romero, D., Stahre, J., Wuest, T., Noran, O., Bernus, P., Fast-Berglund, Å. & Gorecky, D. 2016. Towards an operator 4.0 typology: A human-centric perspective on the fourth industrial revolution technologies. *CIE 2016: 46th International Conferences on Computers and Industrial Engineering*. (October):0–11.

Roof, K. 2018. *Elon Musk says 'humans are underrated,' calls Tesla's 'excessive automation' a 'mistake'* | *TechCrunch*. [Online], Available: https://techcrunch.com/2018/04/13/elon-musk-says-humans-are-underrated-calls-teslas-excessive-automation-a-mistake/ [2018, August 13].

Rother, M. 2009. *Toyota Kata: Managing People for Improvement, Adaptiveness and Superior Results*. US: McGraw-Hill Education. [Online], Available: https://mhebooklibrary.com/doi/book/10.1036/9780071639859.

Ruppert, T., Jaskó, S., Holczinger, T. & Abonyi, J. 2018. Enabling technologies for operator 4.0: A survey. *Applied Sciences (Switzerland)*. 8(9):1–19.

Sanders, M.S. & Mccormick, E.J. 1993. *Human Factors in Engineering and Design*. 7th ed. New York: Mcgraw-Hill. [Online], Available: http://books.google.com/books?id=3d5QPQAACAAJ&pgis=1.

Schroeder, G.N., Steinmetz, C., Pereira, C.E. & Espindola, D.B. 2016. Digital Twin Data Modeling with AutomationML and a Communication Methodology for Data Exchange. *IFAC-PapersOnLine*. 49(30):12–17.

Schweichhart, K. 2016. *Plattform Industrie 4.0 RAMI4.0-a reference framework for digitalisation An Overview*.

Simon, H.A. 1997. Administrative Behavior: A Study of Decision-Making Processes in Administrative Organizations. *The Economic Journal*. 112(480):F386–F388.

Sousa, P., Ramos, C. & Neves,  m J. 2007. Scheduling in holonic manufacturing systems. In *Process Planning and Scheduling for Distributed Manufacturing* (pp. 167-190). Springer, London..

Sparrow, E. 2017. Personal interview. 20 Dec, Abu Dhabi *" The needs of modern composites industries and their expectations of I4.0"*

Sparrow, D.E., Kruger, K. & Basson, A.H., 2019. Architecture for a Human Digital Twin for Human Integration in Industry 4.0 Environments. Proceedings of the 7th *International Conference on Competitive Manufacturing*. Stellenbosch, South Africa.

Sparrow, D., Kruger, K. & Basson, A.H. 2019b. An Abstract Activity Life-Cycle Description for Communication in Cyber-Physical Systems. *Service Orientation in Holonic and Multi-Agent Manufacturing: Proceedings of SOHOMA*

Sparrow, D., Taylor, N., Kruger, K., Basson, A.H. & Bekker, A. 2020. Interfacing with Humans in Factories of the Future; Holonic Interface Services Providing Ambient Intelligence Environments,. in *International Workshop on Service Oriented, Holonic and Multi-Agent Manufacturing Systems for Industry of the Future*.

Stahl, S.M., Davis, R.L., Kim, D.H., Lowe, N.G., Carlson, R.E., Fountain, K. & Grady, M.M. 2010. Play it Again: The Master Psychopharmacology Program as an Example of Interval Learning in Bite-Sized Portions. *CNS Spectr*. 15(8):15.

*Symbolism of Color: Using Color for Meaning*. n.d. [Online], Available: https://www.incredibleart.org/lessons/middle/color2.htm [2019, June 10].

Thalmann, M., Souza, A.S. & Oberauer, K. 2019. How does chunking help working memory? *Journal of Experimental Psychology: Learning, Memory, and Cognition*. 45(1):37–55.

Tharumarajah, A., Wells, A.J. & Nemes, L. 1996. Comparison of the bionic, fractal and holonic manufacturing system concepts. *International Journal of Computer Integrated Manufacturing*. 9(3):217–226.

The Khronos Group Inc. n.d. *glTF Overview*. [Online], Available: https://www.khronos.org/gltf/ [2019a, September 23].

The Khronos Group Inc. n.d. *COLLADA Overview*. [Online], Available: https://www.khronos.org/collada/ [2019b, September 23].

Thiran, J.-Philippe., Marques, Ferran. & Bourlard, H. 2010. *Multimodal signal processing : theory and applications for human-computer interaction*. Academic Press.

Ulrich, E. 2016. *Status Report Industrie 4.0-Technical Assets Basic terminology concepts, life cycles and administration models*. Düsseldorf.

Valckenaers, P. 2019. ARTI Reference Architecture – PROSA Revisited. in *Service Orientation in Holonic and Multi-Agent Manufacturing* Vol. 803. T. Borangiu, D. Trentesaux, A. Thomas, & S. Cavalieri (eds.). (Studies in Computational Intelligence). Cham: Springer International Publishing T. Borangiu, D. Trentesaux, A. Thomas, & S. Cavalieri (eds.). 1–19.

Valckenaers, P. & Van Brussel, H. 2016. *Design for the unexpected : from holonic manufacturing systems towards a humane mechatronics society*. Elsevier.

Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L. & Peeters, P. 1998. Reference Architecture for Holonic Manufacturing Systems. *Computers in Industry*. 37(3):255–274.

VDI. 2015. *Status Report Reference Architecture Model Industrie 4.0 (RAMI4.0)*.

Weiser, M., Gold, R. & Brown, J.S. 1999. The origins of ubiquitous computing research at PARC in the late 1980s. *IBM Systems Journal*. 38(4):693–696.

Woods, D.D., Tittle, J., Feil, M. & Roesler, A. 2004. Envisioning Human–Robot Coordination in Future Operations. *IEEE Transactions on Systems, Man and Cybernetics, Part C (Applications and Reviews)*. 34(2):210–218.

Wrobel, B. & Wolfgang, F. 2004. *Multiple View Geometry in Computer Vision*. University of Oxford: Cambridge University Press. [Online], Available: https://doi.org/10.1017/CBO9780511811685.

Yu, D. 2015. Data Refinement: The Dirty Side of Data Science. [Online], Available: https://insidebigdata.com/2015/06/15/data-refinement-the-dirty-side-of-data-science/ [2020, November 06].

Zaragoza, M.S. & Mitchell, K.J. 1996. Repeated Exposure to Suggestion and the Creation of False Memories. *Journal of Memory and Language*. 35(2):246–260.

# Appendix A.1 - BASE core components detailed internal structure

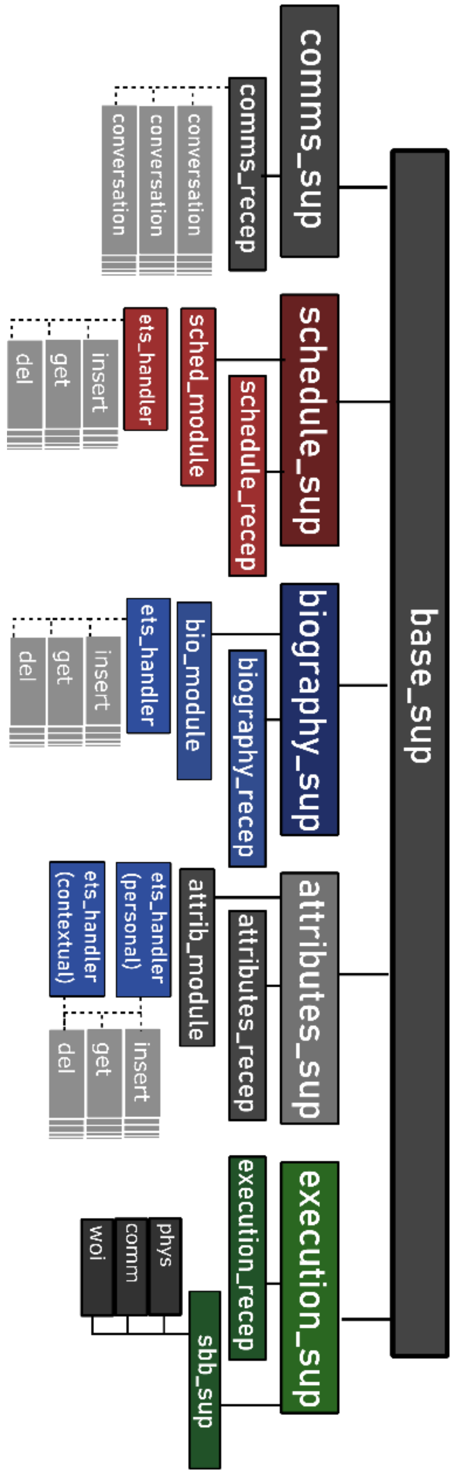



**Figure 67 – A-1 BASE core components supervision tree**

Figure 68 illustrates the internal structure of the Biography core component, the Attributes and Schedule components have similar structures.

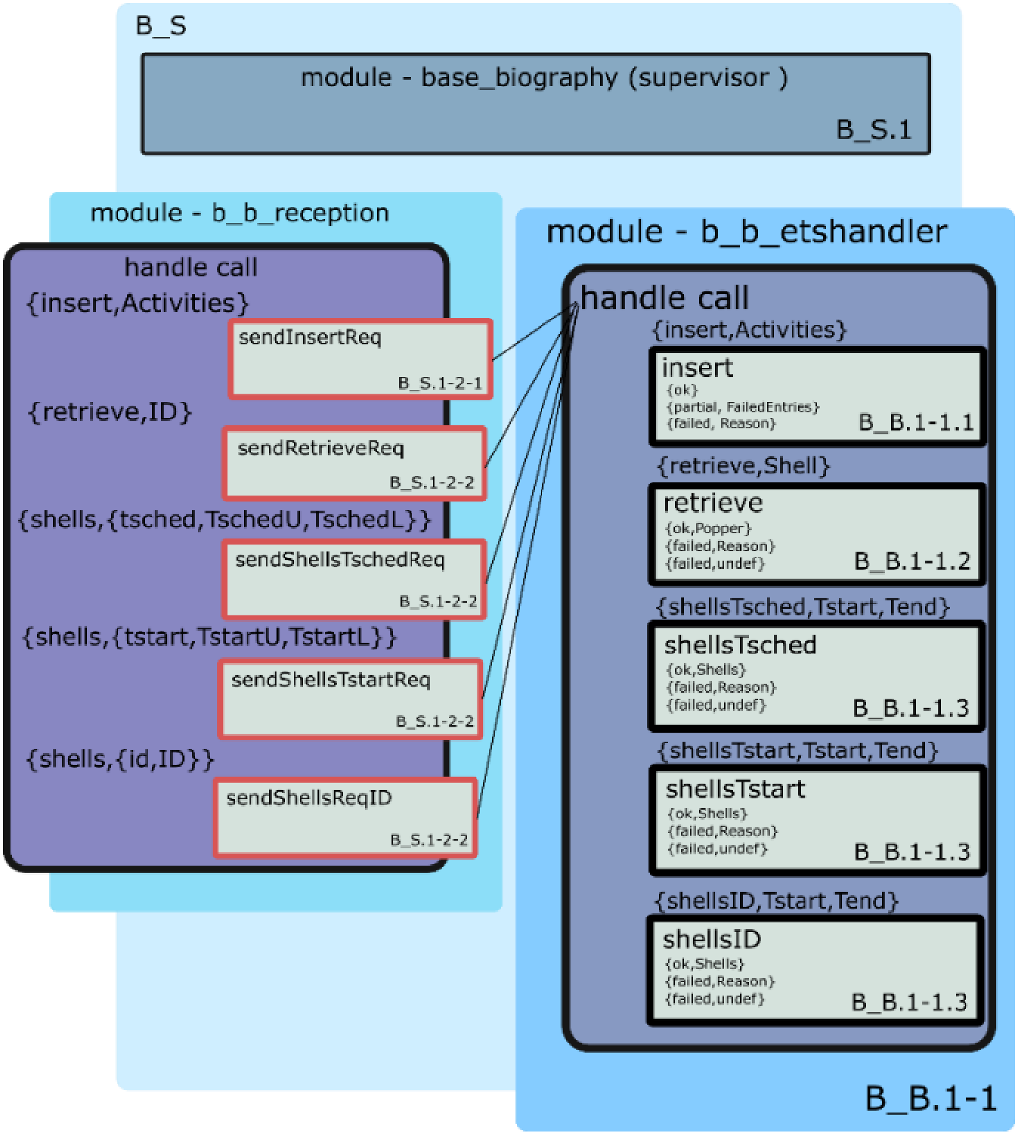



**Figure 68 - The BASE Biography implementation internal structure**

# Appendix A.2 – Observer SV Estimator internal structure

The Execution component's Observer consisted of multiple concurrent State Estimators that updated state variables on the State Blackboard and the exchange of information between external observation services and the internal Execution components are illustrated in Figure 69.

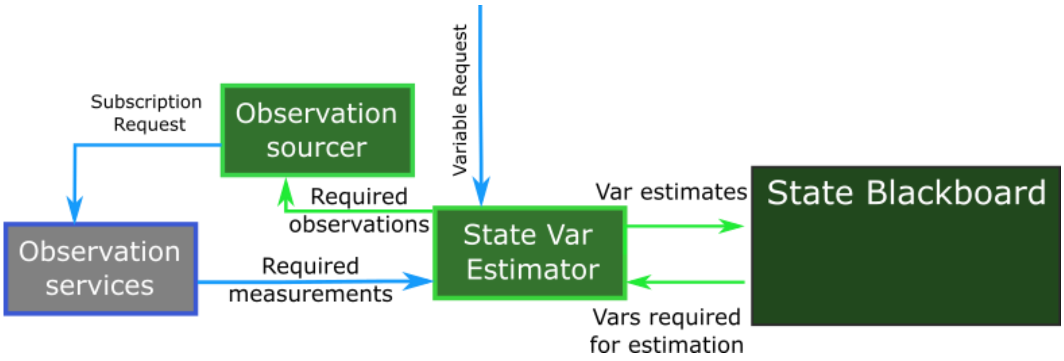



**Figure 69 – State Observer State Variable Estimators message flow diagram**

# Appendix A.3 – State Blackboard internal structure

The internal structure of the State Blackboard is shown in Figure 70. The diagrams were done in a vector graphics format so that components can be expanded to view their internal structure, hence the small size of internal component's text. When viewed in Inkscape these can be expanded.

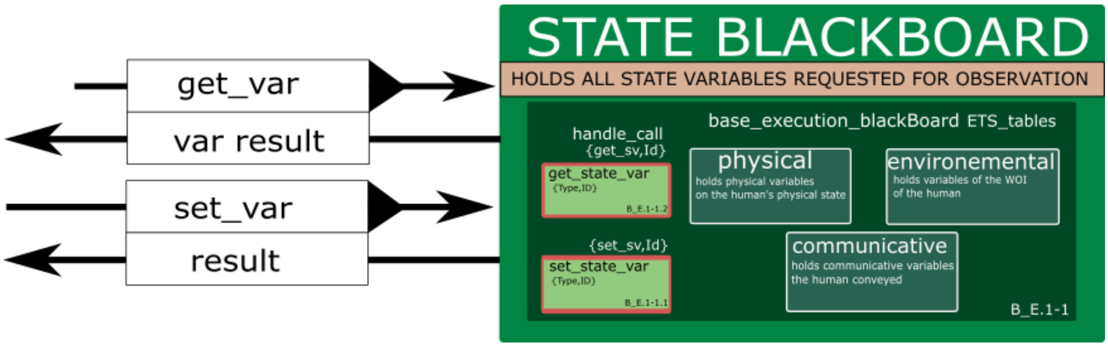



**Figure 70 – State Blackboard internal structure**

The Execution component internal structure is hidden by the API provided through the Reception process. This API is shown in Figure A7

## A.4 BASE and plugins test commands

**Creating a new worker profile with a new BASE BC:**

```
BC = base_business_card:create_std_base_bc(<<"Wandy Carelse">>,
                                    <<"f1fd761e">>,
                                    <<"MOULDER">>).
base:create_profile(BC,"C://BASE/f1fd761e/",[]).
```

**Creating a new activity type in the Layup Activity Holon's Attributes**

```
Steps = [#{<<"name">>=><<"mould_pred">>,<<"dur">>=>200000},
              #{<<"name">>=><<"layup">>,<<"dur">>=>200000},
              #{<<"name">>=><<"baggin">>,<<"dur">>=>200000}].
ReqRes = [<<"MOULDER">>].
Desc1 = <<"Wing fairing Top Outer RH">>,
ExeData1 =
#{<<"steps">>=>Steps,<<"std_w">>=>10800000,<<"req_res">>=>ReqRes,
<<"desc">>=>Desc1}.
lua_type_scheduler:add_lua_type(<<"A0235-00108">>,ExeData1).
```

**Creating a new activity instance on the Layup Activity Holon's Schedule:**

```
Steps = [#{<<"name">>=><<"mould_prep">>,<<"dur">>=>200000},
              #{<<"name">>=><<"layup">>,<<"dur">>=>200000},
              #{<<"name">>=><<"baggin">>,<<"dur">>=>200000}].
ReqRes = [<<"MOULDER">>,<<"MOULD_A0236">>].
Desc = <<"Flap Fairing fwd center LH moulding">>,
ExeData =
#{<<"steps">>=>Steps,<<"std_w">>=>10800000,<<"req_res">>=>ReqRes,
<<"desc">>=>Desc}.
lua_type_scheduler:add_lua_type(<<"A0236-00125_01">>,ExeData).
```

# Appendix B Interface holons detailed implementation descriptions

## B.1 Pose Observation Service Holon

Pose estimation has been developed by several industries including the movie and games industry for capturing actor movements and applying those to 3D characters. These systems usually require the human to wear special reflectors or suits which motion capturing cameras can pick up on. Several other techniques have also become available, many based on artificial intelligence and machine learning. A balance between the accuracy and how invasive the measuring process can be permitted to be must be reached. In the case of a manufacturing environment, it was deemed that a much larger emphasis should be placed on the system staying nearly invisible to the workers, aiming for no special equipment required on their behalf.

A tool the author was familiar with was PoseNet, a TensorFlow application trained to give human pose estimates from camera feeds aimed for browsers and webcams. This meant no special equipment was required and computers with low to moderate processing capabilities could be used. Although the accuracy that PoseNet offered is not as accurate as other options such as Microsoft Kinect, the flexibility to run it as a web application on any device with a camera connected to the network and low development time outweighed the benefits of increased accuracy from the others.

PoseNet outputs the pixel location of human limbs and joints on the image provided, meaning that a 3D pose estimate on the shop floor requires triangulation from 2 or more cameras. The choice of using NodeJS to host a server to which Web Worker devices can connect and run the pose estimation software


**Figure 71 – Example PoseNet output on image obtained from the MLJS5 website**

made this process possible with little effort. The triangulation code was based on lessons from (Wrobel & Wolfgang, 2001).

Any device with HTML5 capabilities and a camera could access the NodeJS server and run the observation service code, giving the server its location and orientation as well as its pose estimate for the human. The server then ran triangulation code on all the pose estimates it received from its web workers, giving a 3D pose estimate. This information is then available to any HRH digital shell that requests to be subscribed to the information stream.

For this case study, two computers with cameras at opposing ends of the lab were logged into the Pose Observer Server (POS) to act as POS Web Workers. Figure 72 shows the entire Pose Observation Interface service.

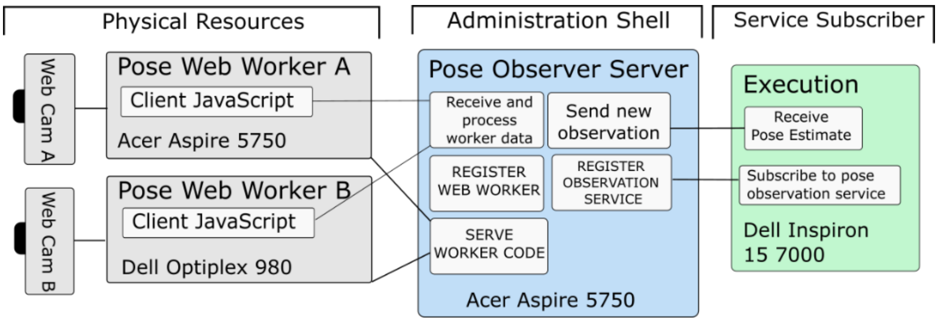



**Figure 72 – Pose Observer structure breakdown**

### B.1.1 Pose Observation Service evaluation:

Three aspects of the Pose Observer were evaluated to determine if it would be effective to the HRH for its evaluation. This evaluation is done apart from the BASE implementation evaluation as it was a tool supporting the architecture evaluation. Figure 73 shows one of the Pose Observation service web workers detecting the

**Figure 73 – View of a web-worker serving the Pose Observer holon**

author at the ply-table

- **Pose estimation:**

The PoseNet code ran on the web workers with webcams along with the custom written triangulation code on the Pose Observer Server successfully provided 3D position data on the joint locations of humans within the camera field of view which covered the workstation.

- **Hardware Redundancy:**

Two different computers running windows 7 and windows 10 respectively, one with a core i3 2nd generation processor and the other with a core i7 9th generation processor were logged into the pose observation server, both starting and running the client code within 7 seconds. An android phone was also used, which successfully opened the onboard camera and ran the same pose estimation code. This proved hardware redundancy provided using HTML5 technology.

- **Reliability:**

Due to open source and mature libraries such as Socket.io being used the connection stability as well as automatic reconnection of web workers to the Pose Observation Server up-time was not a problem. The provided pose estimated saw an accuracy of roughly 20 mm. Centimetre accurate pose estimation exists, but this was enough for proof of concept.

A problem with using PoseNet is that the weights used to load the TensorFlow models are stored on google servers and therefore needs an internet connection to retrieve the weights on start up. This dependency can be solved by storing the weights locally and a request was made on the PoseNet forums which yielded a few suggestions from the community but were left to be implemented on a later iteration.

## B.2 Layup Table state Observation Service Holon

The existing camera feeds used by the pose observation service was used by the ply state observer. The plies used are covered in a red protective film which is removed once a ply needs to be laid. The ply table state observation service was written using image processing based on ply colour. The image processing is done with help from the P5JS JavaScript framework. This observation service is made available to holons in the vicinity. The internal structure of this holon is illustrated in Figure 74

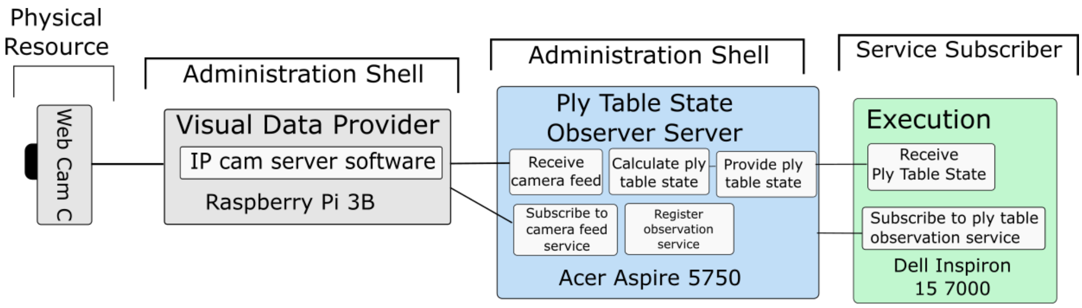



**Figure 74 – Layup table state observer holon internal structure**

# B.3 - Ply projector semiotic service holon

The ply projector semiotic service holon was implemented in the same way as the pose observer holon with a NodeJS server acting as the administration shell connecting to and managing the physical resource and web-worker component. Figure 75 shows the functional flow diagram of the projector's service.

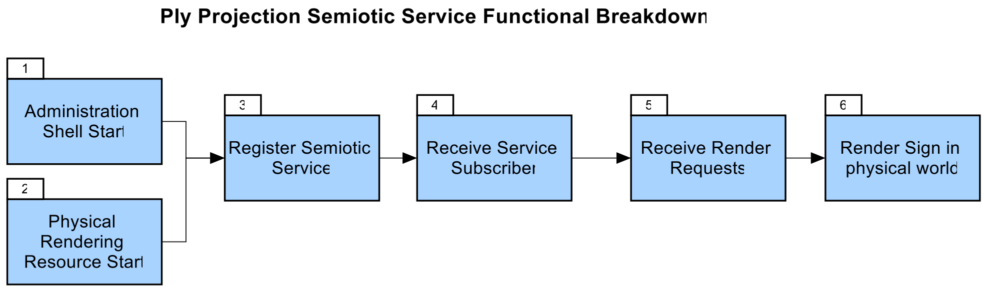



**Figure 75 – Simplified ply projection Semiotic Service functional breakdown**

Figure 76 shows the Execution component interfacing with the ply projector administration shell which has control over the projector.

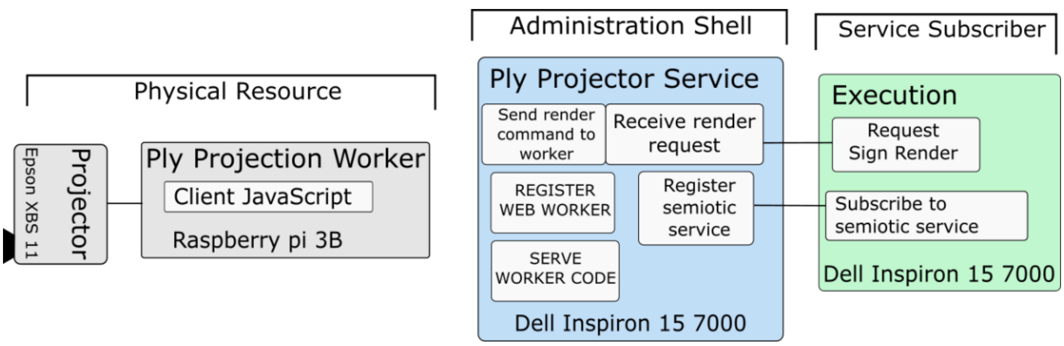



**Figure 76 – Internal structure of the ply projector semiotic service holon**


**Figure 77 – Projector mounted over the layup station connected to a Raspberry Pi 3 to act as the physical rendering component of the Ply Projector semiotic service holon**

# Appendix C - LUA execution plugin details

Figure 78 shows the LUA steps planned out along with the required observations, instructions to be rendered (imperatives) and the preferred interfaces.


**Figure 78 – LUA action plan along with observations and imperatives to be rendered**

During the example Unpack Plies state shown in Figure C-1, the execution tools execute an NUE step by requesting imperatives be rendered to the human and monitoring for key WOI and HRH variables. A sample of the algorithmic flow is given in Figure 79 showing the Signs and Monitors given to the Execution AHP.

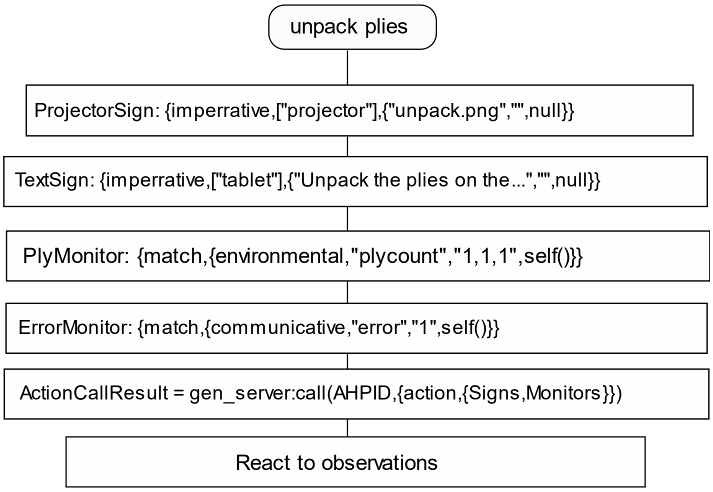



**Figure 79 – A simplified example of the Layup Activity state machine process flow during a state transition**

# Appendix D - Tablet interface development for case study 1

The human needs to take part in a contract net protocol to prove that they can integrate with the holonic communication chain. It was chosen to have a tablet interface service offering bi-directional communication and direct interaction with the HRH schedule. The authors had extensive experience in developing 2D games on android with the LibGDX framework, and it was chosen to use this framework and the flexibility it provides to develop the tablet interface. This allowed any android device to become to register the required semiotic and observational services required for RFP and error reporting dialogs. The Tablet interface service will be the most information rich and needs to serve the following functions:

## D.1 Tablet interface requirements

1. Provide means of submitting a proposal for a requested activity or refusing the proposal.
2. Provide a means of rendering text signs from the HRH administration shell or external holons
3. Provide means for the human to indicate an error occurred in the process so that the administration shell can appropriate steps
4. Provide a means of capturing an image as well as a text description of the error from the human to be logged as part of the current activity's execution data block.
5. Allow the user to confirm the start of an activity or cancel it

## D.2 Tablet interface UI design

Figure 80 shows the UI design for the tablet interface where the user will receive new activity requests on the side, which can be dragged and dropped onto the schedule timeline on the right.

User Interface (UI) and User Experience (UX) game design principles were initially used to design the interface. Although the interface could be professionally designed, the feedback from the testing session deemed the interface to be intuitive and easy to work with, which is enough for the purposes of this case study.

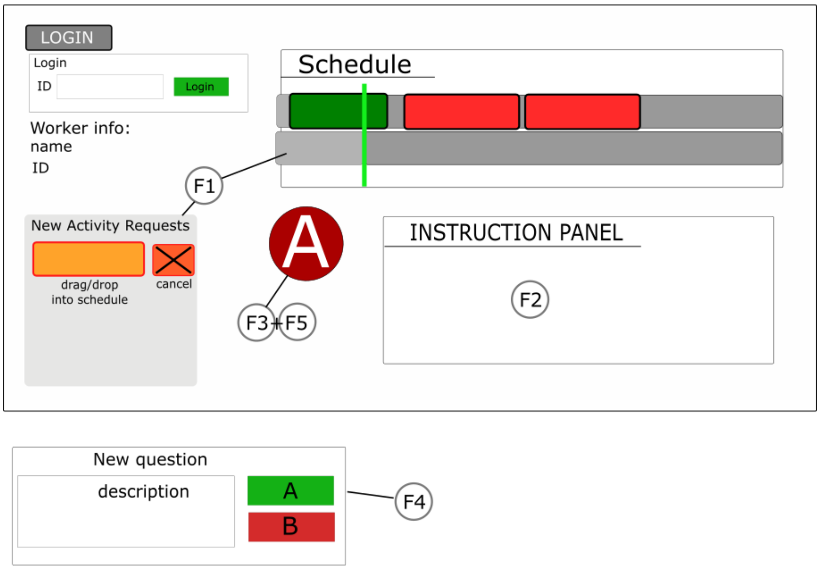



**Figure 80 – Tablet interface UI design with function labels**

## D.3 Tablet interface user Schedule management component

Using the value for the average duration of the requested activity as stored in the administration shell's Attributes component, the length of the activity is reflected on the schedule timeline. Once the activity is placed, the interface writes the proposed activity start time to the Communicative State Variables (CSVs) which the schedule plugins and execution plugins can then react to. Figure 81 labelled each UI component with the function that it fulfils.

Figure 81 shows a tablet interface screenshot with an accepted layup activity as well as its automatically added pose logging activity on the HRH schedule. No text-based signs are rendered, and no other activity requests have been made as can be seen on the other UI components


**Figure 81 – Tablet interface Schedule screenshot.**

Figure 83 and 84 shows a diagram made during the planning and design phase of the tablet interface. All communication channels and the components responsible for handling them were laid out before implementation.

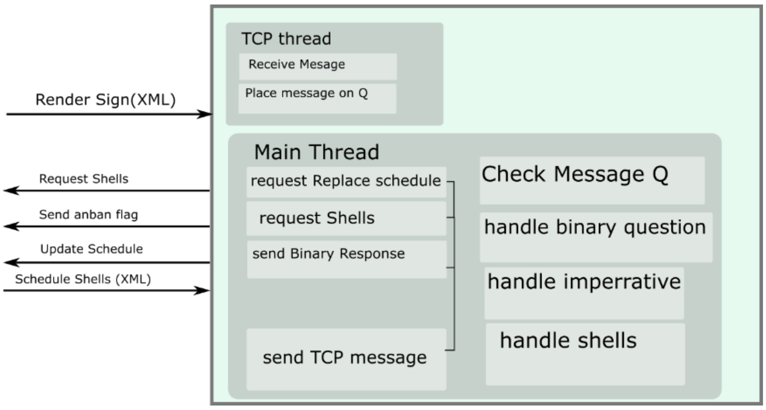



**Figure 83 – GUI internal component planning and message handling from the tablet java code side**

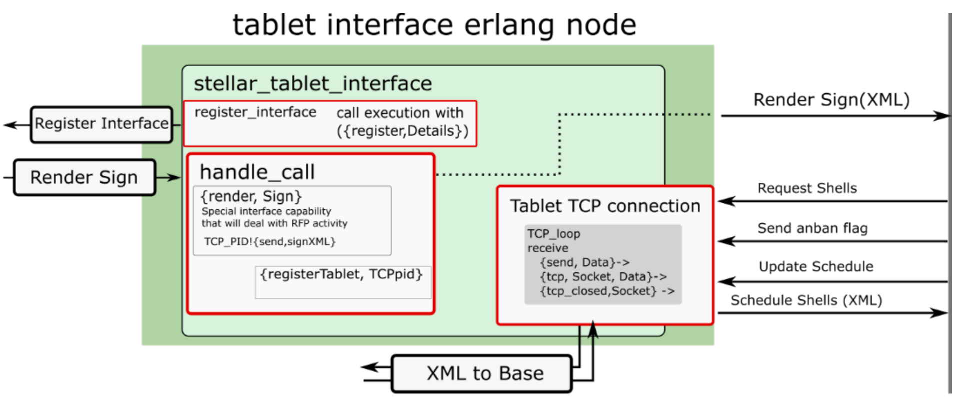



**Figure 82 – GUI internal component planning and message handling from the erlang node side.**

# Appendix E – Photo sequence of layup activity and rendered signs

Figure 84 shows a sample of a few process steps along with the rendered instruction through the ply projector semiotic service.

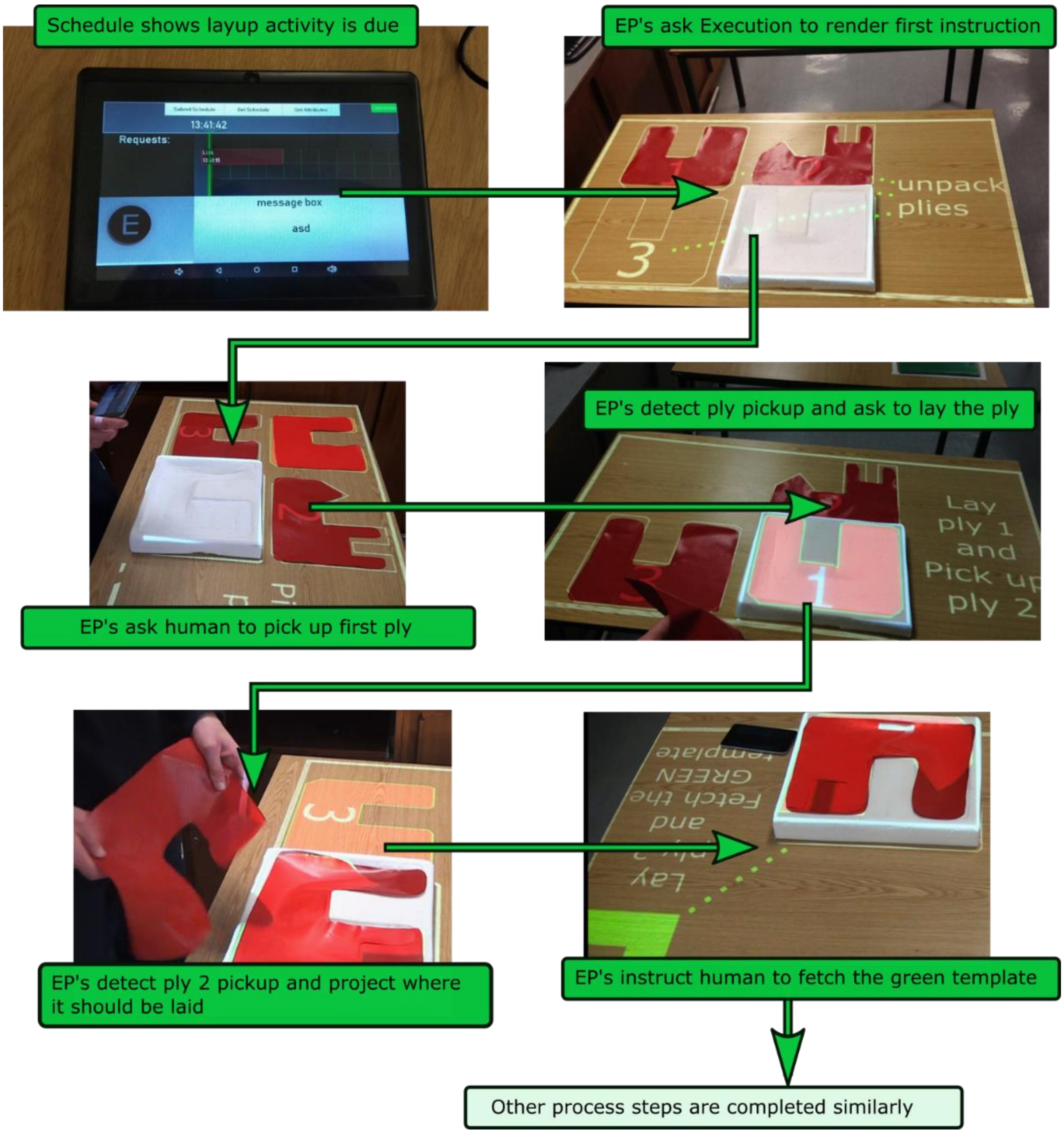



**Figure 84 – A sequence of rendered instructions triggered by the Execution Plugins monitoring the human and his WOI state.**

# Appendix F - Evaluation of secondary ALPs

**Table 39 – ALP_1.2 Effectiveness scoring for error reporting**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 0 sec | ~ 20 sec | ~5 sec | ~25 sec |
| | Bandwidth | - | ~ 4 MB/s | ~5 MB/s | ~5 Mb/s |
| Accuracy | Degradation | - | 0%/min | 0%/min | 0%/min |
| | Compression | - | ~10% | ~10% | ~10% |
| | Success rate | 100% | | | 100% |
| | Correct destination | | 100% | | 100% |
| | Data recentness | | | 2/3 | 2/3 |
| | Size limit | - | ~33 bytes | ~33 bytes | ~33 bytes |
| Visibility | Availability | 100% | ~30% | ~14% | ~10% |
| | Traceability | 1/3 | 1/3 | 1/3 | 1/3 |

**Table 40 – ALP_1.3 Effectiveness scoring for error reporting**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 0 | ~ 2 s | ~0.5 s | ~2.5 sec |
| | Bandwidth | | ~ 5 Mb/s | ~ 5 Mb/s | ~ 5 Mb/s |
| Accuracy | Degradation | | 0 %/min | 0 %/min | 0 %/min |
| | Compression | | 10% | 10% | 10% |
| | Success rate | ~99% | | | ~99% |
| | Correct destination | | 100% | | ~99% |
| | Data recentness | | | 3/3 | 3/3 |
| | Size limit | | 16+ MB | 16+ MB | 16+MB |
| Visibility | Availability | | ~99% | ~99% | ~99% |
| | Traceability | | 3/3 | 3/3 | 3/3 |

**Table 41 – ALP_2.2 Effectiveness scoring for obtaining activity state information**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~ 25 min | ~0.4 min | ~13 min |
| | Bandwidth | ~ 5 Mb/s | 5 MB/s | ~5 Mb/s |
| Accuracy | Degradation | 0%/min | 0 %/min | ~0%/min |
| | Compression | ~10 % | ~ 10% | ~10% |
| | Success rate | | | ~99 % |
| | Correct destination | 99 % | | ~99 % |
| | Data recentness | | 1/3 | 1/3 |
| | Size limit | ~33 bytes | 5 MB | 5 MB |
| Visibility | availability | ~5% | ~14% | ~10% |
| | traceability | 1/3 | 1/3 | 1/3 |

**Table 42 – ALP_2.3 Effectiveness scoring for obtaining activity state information**

| Characteristic | Property | Transport | Store | Combination |
|---|---|---|---|---|
| Responsiveness | Time | ~ 2 s | ~0.5 s | ~2.5 sec |
| | Bandwidth | ~ 5 Mb/s | ~ 5 Mb/s | ~ 5 Mb/s |
| Accuracy | Degradation | 0%/min | 0%/min | ~0%/min |
| | Compression | 10% | 0% | ~10% |
| | Success rate | | | ~100% |
| | Correct destination | 100% | | ~100% |
| | Data recentness | | 3/3 | 3/3 |
| | Size limit | 16+ MB | 16+ MB | 16+ MB |
| Visibility | Availability | ~99% | ~99% | ~99% |
| | Traceability | 3/3 | 3/3 | 3/3 |

**Table 43 - ALP_3.2 Effectiveness scoring for obtaining Standard Work**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 0 | 2 s | ~0.5 s | ~2.5 s |
| | Bandwidth | - | 5 MB/s | ~5 MB/s | ~5 MB/s |
| Accuracy | Degradation | - | 0 Mb/s | 0 Mb/s | ~0 MB/s |
| | Compression | - | ~10 % | ~10 % | ~10 % |
| | Success rate | ~99% | | | ~99 % |
| | Correct destination | - | ~99% | | ~99 % |
| | Data recentness | - | | 3/3 | 3/3 |
| | Size limit | - | ~10 Mb | 16+ Mb | 16+ MB |
| Visibility | Function availability | ~99 % | ~99 % | ~99 % | ~99% |
| | Function traceability | - | 3/3 | 3/3 | 3/3 |

**Table 44 – ALP_4.2 Effectiveness scoring on tooling and maintenance information**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 3 min | 5 sec | ~ 10 min | ~13 min |
| | Bandwidth | - | 5 Mb/s | ~7.52b/s | ~5 Mb/s |
| Accuracy | Degradation | - | ~0%/h | ~ 2%/min | ~2%/min |
| | Compression | - | ~10% | ~ 50% | ~50 % |
| | Success rate | ~99 % | | | ~99 % |
| | Correct destination | | 30% | | ~99 % |
| | Data recentness | | | 2/3 | 1/3 |
| | Size limit | - | 5 MB | 5 MB | 5 MB |
| Visibility | Function availability | - | ~14% | ~14% | ~14% |
| | Function traceability | - | 1/3 | 2/3 | 1/3 |

**Table 45 – ALP_4.3 effectiveness scoring on tooling and maintenance information**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 0 | 0.5 s | ~2 s | ~2.5 s |
| | Bandwidth | - | 10Mb/s | ~1kb/s | 7 bytes/s |
| Accuracy | Degradation | - | 0 Mb/s | 0 Mb/s | 0 Mb/s |
| | Compression | - | 10 % | 10 % | 10 % |
| | Success rate | 100 % | | | 100 % |
| | Correct destination | - | 100 % | | 100 % |
| | Data recentness | - | | 3/3 | 3/3 |
| | Size limit | - | ~10 Mb | 16+ Mb | 16+ Mb |
| Visibility | Function availability | 100 % | 99 % | ~99 % | 99% |
| | Function traceability | - | 3/3 | 3/3 | 3/3 |

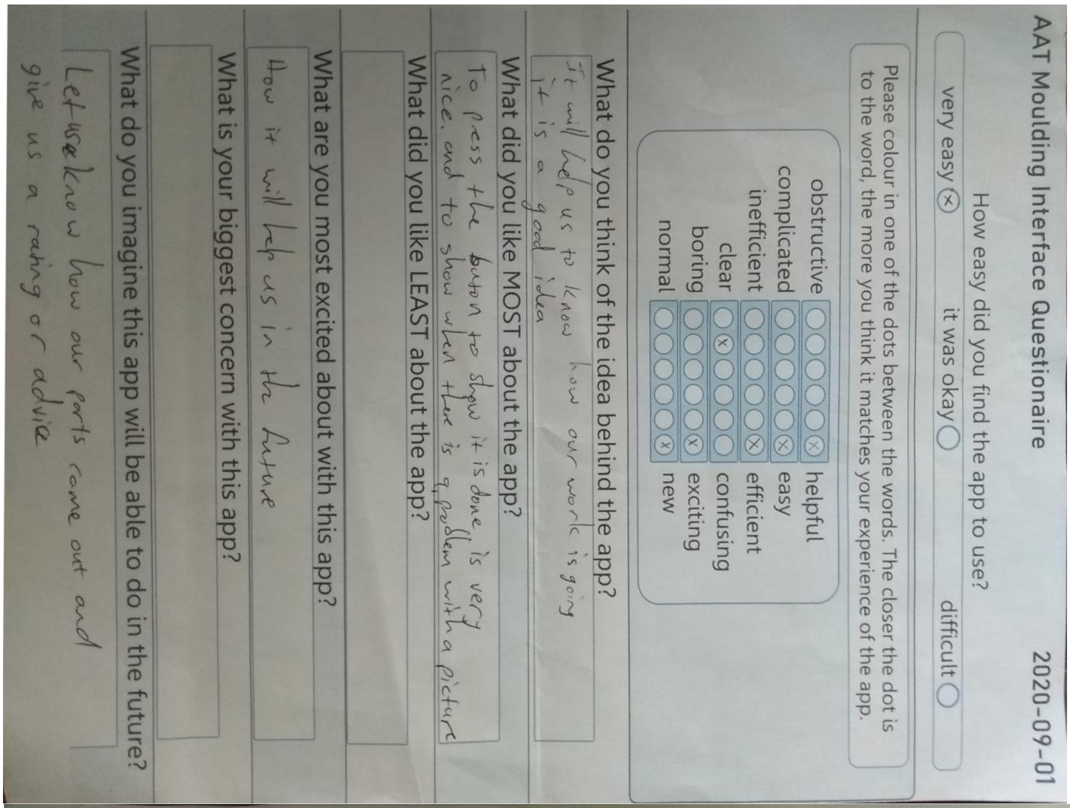
**Table 46 – ALP_5.2 Evaluation (worker feedback)**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 3 min | ~25s min | ~10 min | ~13 min |
| | Bandwidth | - | 5 MB/s | 10 MB/s | ~5 MB/s |
| Accuracy | Degradation | - | | 2 %/min | ~2 %/min |
| | Compression | - | 50% | 50% | ~25 % |
| | Success rate | 10 % | | | ~10 % |
| | Correct destination | | 100 % | | ~99 % |
| | Data recentness | | | 1/3 | 1/3 |
| | Size limit | - | 33 bytes | > 10Mb/s | ~33 bytes |
| Visibility | Function availability | ~100 % | ~14% | ~14% | ~14% |
| | Function traceability | 1/3 | 1/3 | 1/3 | 1/3 |

**Table 47 – ALP_5.3 Evaluation (worker feedback)**

| Characteristic | Property | Locate | Transport | Store | Combination |
|---|---|---|---|---|---|
| Responsiveness | Time | 0 | 0.5 s | ~2 s | ~2.5 s |
| | Bandwidth | - | ~7 bytes/s | ~10 MB/s | ~7 bytes/s |
| Accuracy | Degradation | - | 0 Mb/s | 0 Mb/s | ~0 Mb/s |
| | Compression | - | 10 % | 10 % | ~10 % |
| | Success rate | ~99% | | | ~99% |
| | Correct destination | - | ~99% | | ~99% |
| | Data recentness | - | | 3/3 | 3/3 |
| | Size limit | - | ~10 Mb | 33 bytes | 33 bytes |
| Visibility | Function availability | ~60 % | ~14 % | ~14 % | ~14 % |
| | Function traceability | - | 3/3 | 3/3 | 3/3 |

# Appendix G - Worker questionnaires



AAT Moulding Interface Questionnaire 2020-09-01

How easy did you find the app to use?

very easy it was okay difficult

Please colour in one of the dots between the words. The closer the dot is to the word, the more you think it matches your experience of the app.

| | |
|---|---|
| obstructive | helpful |
| complicated | easy |
| inefficient | efficient |
| clear | confusing |
| boring | exciting |
| normal | new |

What do you think of the idea behind the app?

It will help us to know how our work is going it is a good idea

What did you like MOST about the app?

To press the button to show it is done is very nice, and to show when there is a problem with a picture

What did you like LEAST about the app?

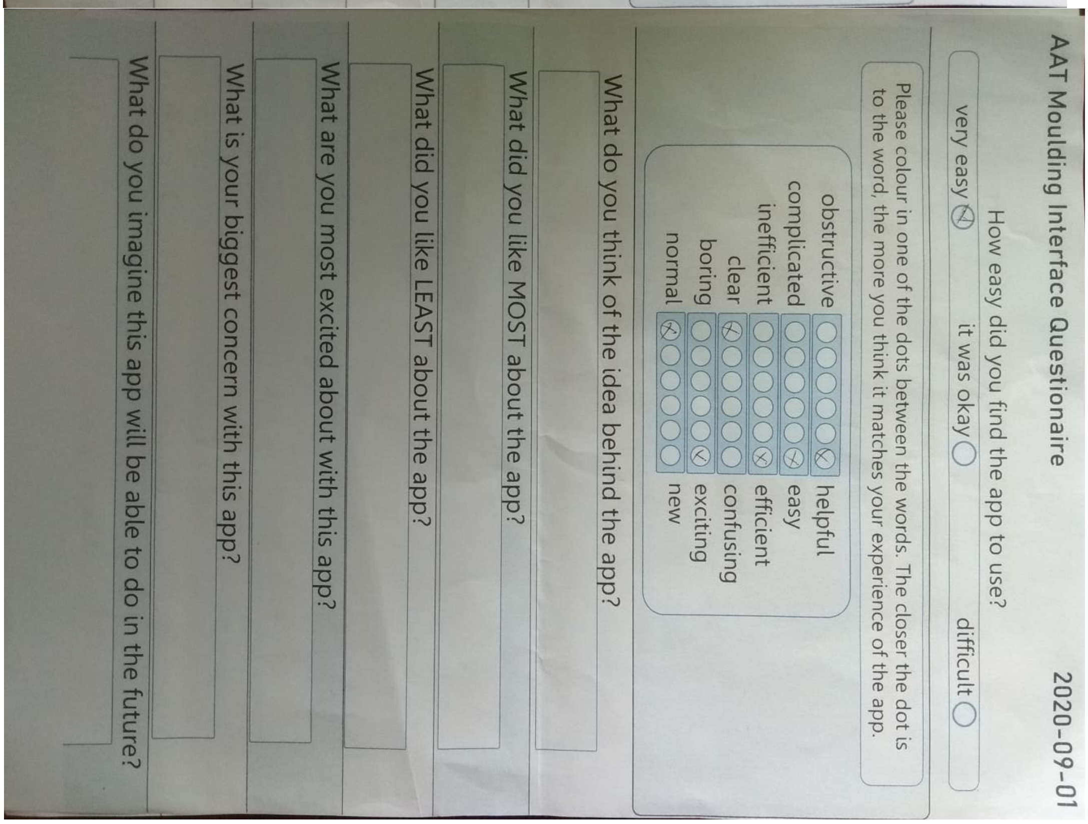
What are you most excited about with this app?

How it will help us in the future

What is your biggest concern with this app?

What do you imagine this app will be able to do in the future?

Let us know how our parts come out and give us a rating or advice



AAT Moulding Interface Questionnaire 2020-09-01

How easy did you find the app to use?

very easy it was okay difficult

Please colour in one of the dots between the words. The closer the dot is to the word, the more you think it matches your experience of the app.

| | |
|---|---|
| obstructive | helpful |
| complicated | easy |
| inefficient | efficient |
| clear | confusing |
| boring | exciting |
| normal | new |

What do you think of the idea behind the app?

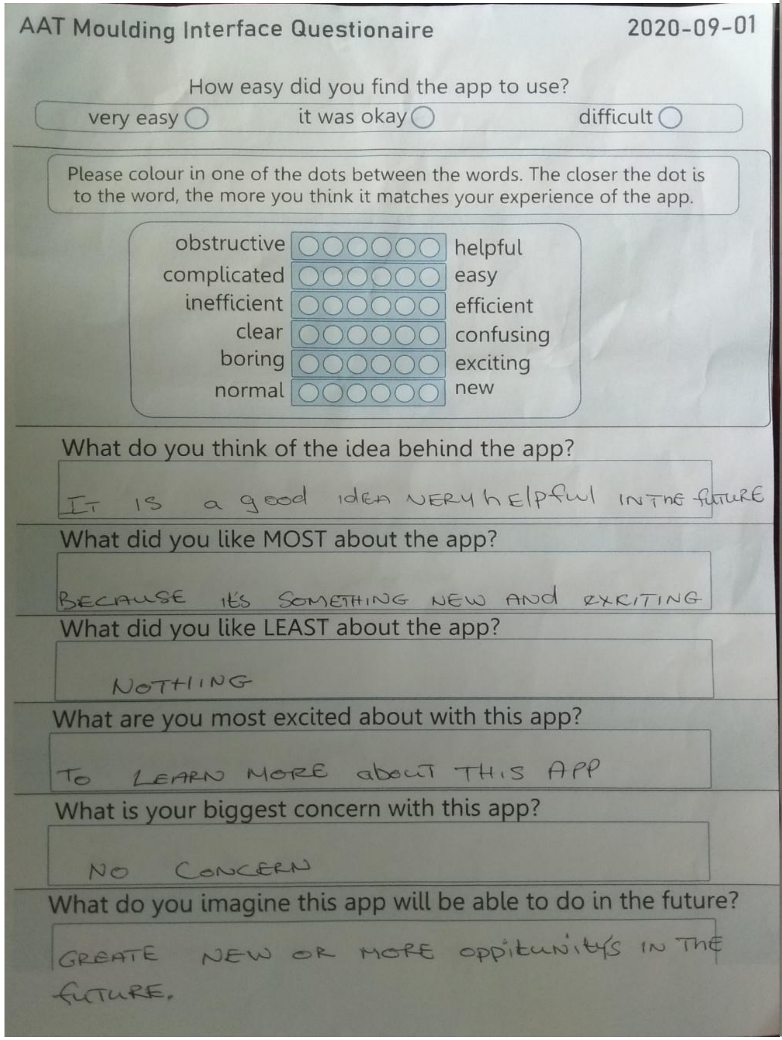
What did you like MOST about the app?

What did you like LEAST about the app?

What are you most excited about with this app?

What is your biggest concern with this app?

What do you imagine this app will be able to do in the future?



AAT Moulding Interface Questionaire 2020-09-01

How easy did you find the app to use?

very easy ◯ it was okay ◯ difficult ◯

Please colour in one of the dots between the words. The closer the dot is to the word, the more you think it matches your experience of the app.

| | | |
|---|---|---|
| obstructive | ◯◯◯◯◯◯ | helpful |
| complicated | ◯◯◯◯◯◯ | easy |
| inefficient | ◯◯◯◯◯◯ | efficient |
| clear | ◯◯◯◯◯◯ | confusing |
| boring | ◯◯◯◯◯◯ | exciting |
| normal | ◯◯◯◯◯◯ | new |

What do you think of the idea behind the app?

IT IS a good idea VERY helpful IN THE future

What did you like MOST about the app?

BECAUSE it's SOMETHING NEW AND EXCITING

What did you like LEAST about the app?

NOTHING

What are you most excited about with this app?

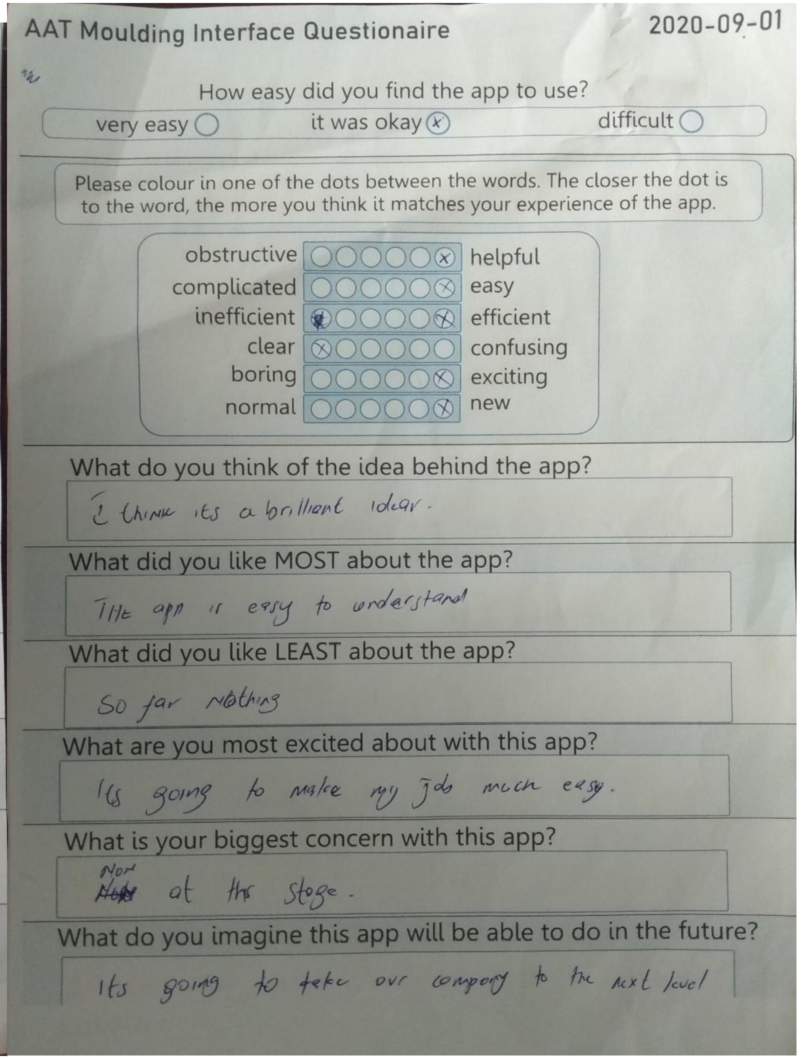
TO LEARN MORE about THIS APP

What is your biggest concern with this app?

NO CONCERN

What do you imagine this app will be able to do in the future?

GREATE NEW OR MORE oppitunitys IN THE future.



AAT Moulding Interface Questionaire 2020-09-01

How easy did you find the app to use?

very easy ◯ it was okay ⊗ difficult ◯

Please colour in one of the dots between the words. The closer the dot is to the word, the more you think it matches your experience of the app.

| | | |
|---|---|---|
| obstructive | ◯◯◯◯◯⊗ | helpful |
| complicated | ◯◯◯◯◯⊗ | easy |
| inefficient | ●◯◯◯◯⊗ | efficient |
| clear | ⊗◯◯◯◯◯ | confusing |
| boring | ◯◯◯◯◯⊗ | exciting |
| normal | ◯◯◯◯◯⊗ | new |

What do you think of the idea behind the app?

I think its a brilliant idear.

What did you like MOST about the app?

The app is easy to understand

What did you like LEAST about the app?

So far nothing

What are you most excited about with this app?

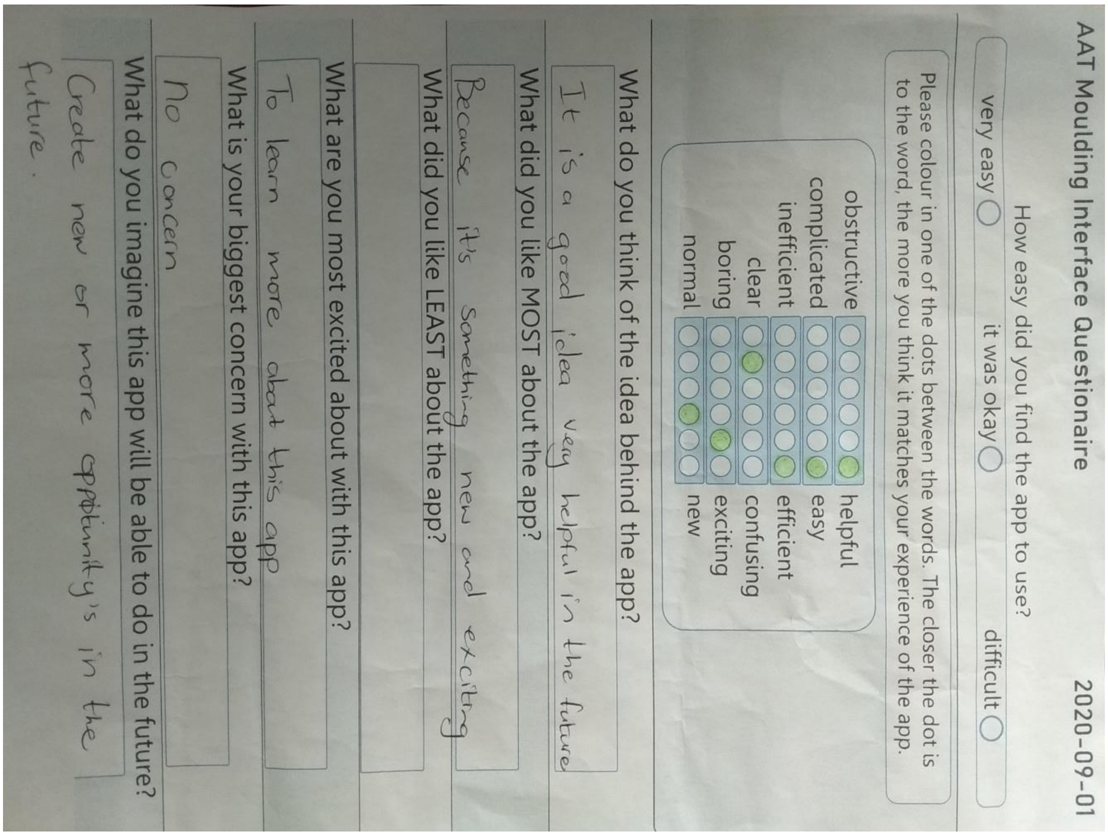
Its going to make my job much easy.

What is your biggest concern with this app?

~~Non~~ None at the stage.

What do you imagine this app will be able to do in the future?

Its going to take our company to the next level



AAT Moulding Interface Questionaire 2020-09-01

How easy did you find the app to use?

very easy ◯ it was okay ◯ difficult ◯

Please colour in one of the dots between the words. The closer the dot is to the word, the more you think it matches your experience of the app.

| | | |
|---|---|---|
| obstructive | ◯◯◯◯◯● | helpful |
| complicated | ◯◯◯◯◯● | easy |
| inefficient | ◯◯◯◯◯● | efficient |
| clear | ◯●◯◯◯◯ | confusing |
| boring | ◯◯◯◯●◯ | exciting |
| normal | ◯◯◯●◯◯ | new |

What do you think of the idea behind the app?

It is a good idea very helpful in the future

What did you like MOST about the app?

Because it's something new and exciting

What did you like LEAST about the app?

What are you most excited about with this app?

To learn more about this app

What is your biggest concern with this app?

no concern

What do you imagine this app will be able to do in the future?

Create new or more oppotunity's in the future.

# Appendix H – Worker interface schematic case study 2

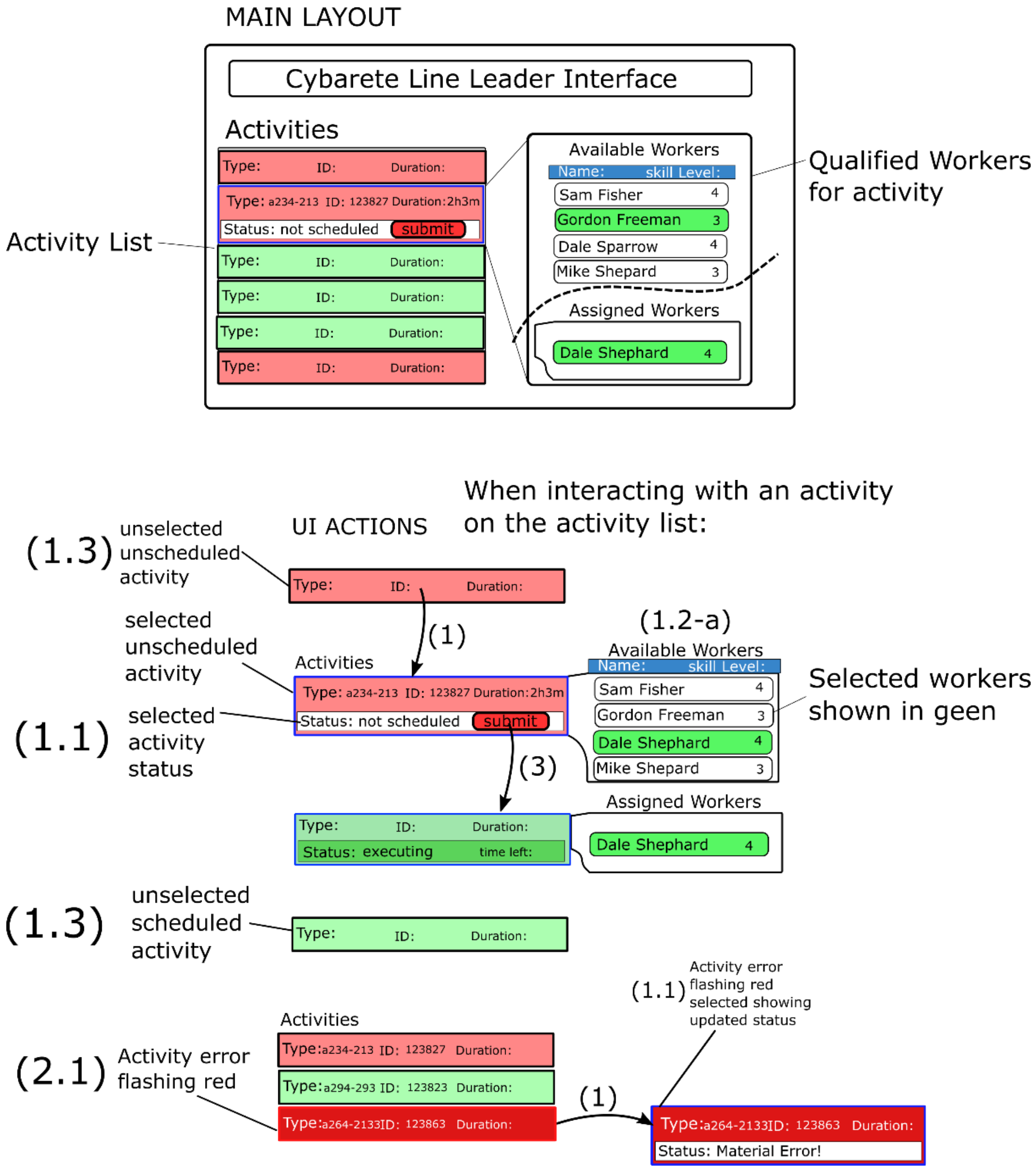



**Figure 85 – Moulding worker interface schematic case study 2**

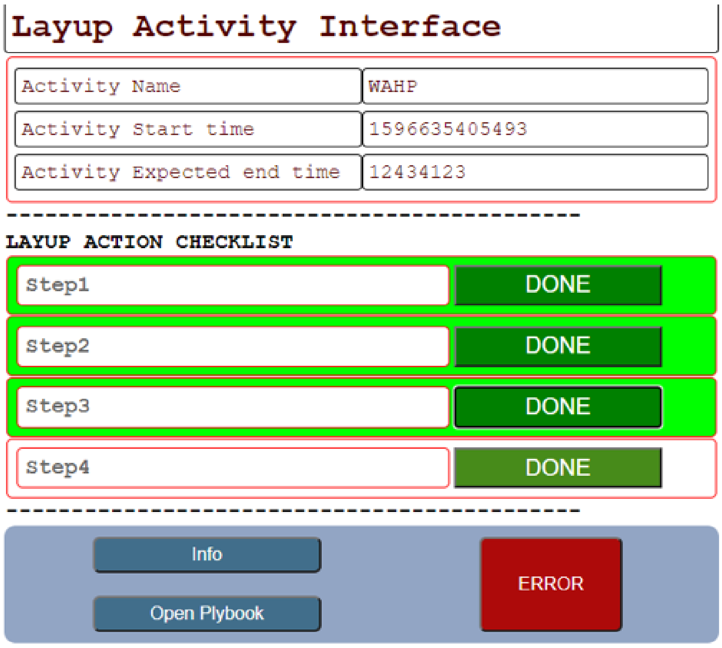



**Figure 86 – Screenshot of moulding worker interface**

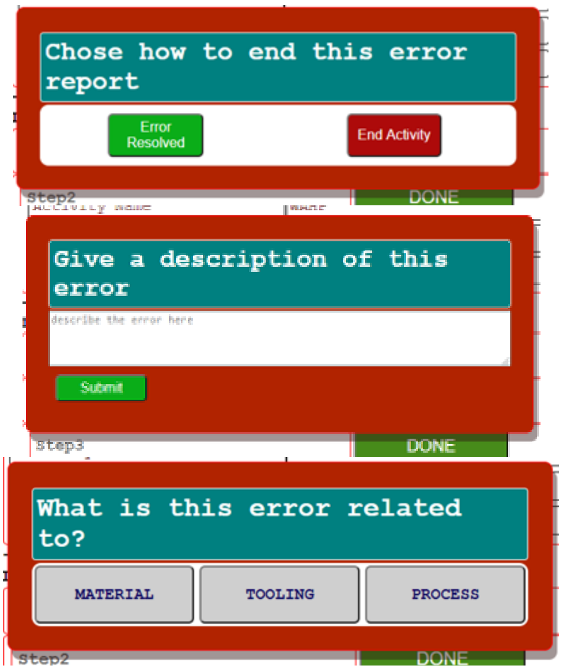



**Figure 87 – Error reporting dialog served on moulding worker interface**

# Appendix I - Review from industry partner liaison on case study 1

1) Does the BASE implementation:

- Improved traceability of process deviations? – Yes, by providing stored records of actions performed.
- Improve the communication chain between the shop floor and engineering? – Yes, by providing records in an easily understandable format (photographic and/or video data) linked to a specific Serial Number, time of execution etc. to provide clarity and context.
- Improve real-time scheduling decisions? Potentially yes. On a limited scale (eg. kit-cutting & moulding) I see practical opportunities. On a plant wide scale (running 24/7) it becomes a very complex problem. Considering our experience in rolling out an ERP, it is a major undertaking which is difficult to comment on with my current knowledge of the system.

2) Was the request for automating Standard Work with time and motion analysis shown to be feasible using BASE? Yes, it looked feasible for the moulding station application which was the focus of the demonstration. It can probably also be adapted to other work areas.

3) Would you say an implementation of the BASE architecture would add value to a manufacturing environment such as yours? Yes

4) Would you be interested in further developing the capabilities of the BASE implementation for your use? Yes, I think a good next step would be to do an on-site demo with more stake-holders from various departments present. A successful demo could very well lead to the definition of a pilot project.

5) Do you see an administration shell such as BASE to be the future of industry 4.0 factory worker integration? There is a real interface gap which needs to be solved. My impression is that such a shell could very well be a key element in solving this problem.

6) Do you have any other comments or suggestions regarding what you saw at the demonstration? Often operators (eg. moulders) mould a part on one shift, and then the parts are de-moulded and evaluated on a next shift, continuing downstream into assembly. Mostly the moulders do not see or receive feedback on what the result of their work was (unless there is a serious problem). Feedback to operators on the quality of their work (also when the work is of the required standard, or improving), could drive continuous improvement and also help worker motivation.