LeArning and robuSt deciSIon SupporT systems for agile mANufacTuring environments

Project Acronym:

ASSISTANT

Grant agreement no: 101000165

Deliverable no. and title	**D5.1 - Requirements for multi-level reconfigurable manufacturing execution**	
Work package	WP5	Real-time control and actuation
Task	T5.1	Assessment of key elements and multi-level requirements for reconfigurable manufacturing execution.
Lead contractor	Institut Mines-Telecom (IMT) Alexandre Dolgui, mailto: alexandre.dolgui@imt-atlantique.fr	
Deliverable responsible	University of Patras, Laboratory for Manufacturing Systems and Automation (LMS)	
Version number	v1.0	
Date	23/06/2021	
Status	Final	
Dissemination level	Public (PU)	

Authors

Participant no.	Part. short name	Author name	Chapter(s)
1	IMT	Institut Mines-Telecom	-
2	UCC	University College Cork	3,4
3	LMS	University of Patras - Laboratory for Manufacturing Systems and Automation	All
4	FLM	Flanders Make	4,5
5	TUM	Technical University of Munich	-
6	BITI	Biti Innovations AB	3
7	SAG	SIEMENS AG	-
8	INTRA	INTRA SOFT	3,4
9	AC	Atlas Copco	5
10	SE	SIEMENS Energy	-
11	PSA	Groupe PSA	5

Document History

Version	Date	Author name	Reason
v0.1	04.12.2020	IMT	Initial Template
v0.2	06.04.2021	LMS	Outline of the document
v0.3	06.05.2021	WP5 partners	First version with all inputs
v0.4	27.05.2021	WP5 partners	Second version of the full draft
v0.5	03.06.2021	LMS	Full draft ready for internal review
v0.6	11.06.2021	FLM	First review comments
v0.7	15.06.2021	TUM	Second review comments
v0.8	17.06.2021	UCC, BITI	Update on the comments
v0.9	22.06.2021	LMS, INTRA, FLM	Update on the comments
v0.9.1	23.06.2021	WP5 partners	Final version of the deliverable
v0.9.2	28.06.2021	WP5 partners	Minor modifications
v1.0	28.06.2021	Félicien BARHEBWA MUSHAMUKA	Final editing and preparation for submission

Publishable Executive Summary

The D5.1 of the ASSISTANT project outlines the specifications and requirements of WP5's results and in particular its primary outcome: the real-time control and actuation of a production system. The activities performed under Task 5.1 of WP5 are documented in this report. WP5 will pursue different objectives related to real-time actuation and puts the human at the core of the technology. The overall architecture of this work package is composed of four core technologies: a) execution control, b) digital twin c) human-robot collaboration d) workers assistive interfaces. The deliverable has been organized in accordance with this architecture to first present the state of the art of these four technologies and then to further analyze the requirements of the anticipated ASSISTANT solution.

About the ASSISTANT Project

The ASSISTANT project is a European Horizon 2020 Research and Innovation project developing AI-based tools for adaptive manufacturing. The project's goal is to develop next-generation artificial intelligence tools and techniques to boost the European manufacturing industry's cost efficiency, production performance, and versatility. The ASSISTANT is organized in nine work packages spanning from project management to technical development and integration of project results in real-world use cases. This report outlines the context of task 5.1 of work package 5, identifies requirements, and leads into key technical design decisions. It will be followed by three further deliverables that will better illustrate the activities of WP5 during the project.

Table of contents

List of figures

List of tables

1. Introduction

1.1 Objective and general approach

The industry has taken a big leap forward by placing a human in the center of interest by turning the working areas into a collaborative environment between operators and robots. Considering the fluctuating market that constantly requests new and customizes products, flexibility has become a key to the sustainability of EU manufacturing companies. Production and research engineers have resorted to the concept of human-robot collaboration (HRC) to comply with the flexibility requirements. The desired result from creating human-robot collaborative environments is for operators to be able to coexist and share tasks with robots in the same working area. In these environments, human behavior is a major uncertainty factor that can affect an operator’s safety and production’s execution status. The creation of a digital twin including the whole workstation area, the operators, and the procedures that take part in there, is a way to design and integrate collaborative systems using a virtual space. The central challenges that WP5 is responsible to address, are the real-time actuation of several components and the placement of humans at the core of the production. Flexible resource handling methods, as well as robot behavior adaptation methods for ensuring safe and efficient Human-Robot collaboration (HRC), will be developed. WP5 will also focus on developments regarding wearable devices that will help human task tracking and human intention estimation techniques to provide a collaborative human-robot environment. The whole implementation of WP5 is based on the integration of some basic technologies.

As depicted in Figure 1, WP5 consists of four main tasks that provide four different outcomes related to real-time actuation as well as placing the human as the core technology. To begin with, an important issue to be solved in production is the direct interaction between the operator or the production manager and the system. For that purpose, Task 5.2 develops the human side interfaces that will accomplish the human centric interaction with the system.

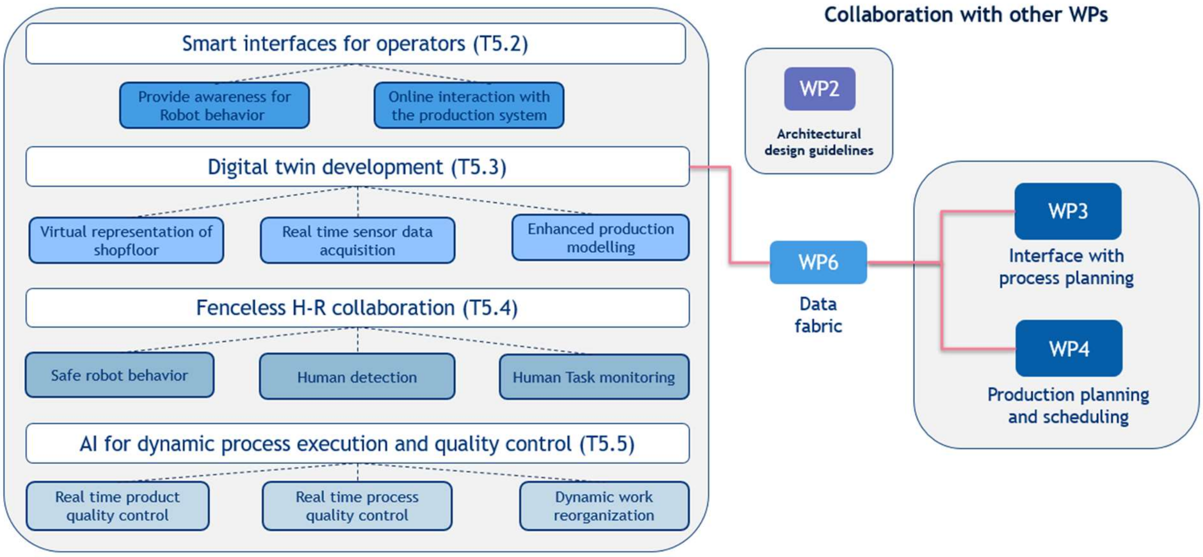

Figure 1: WP5 overall approach

There are many cases in which the operator needs to be informed during the production. The operator is informed about the execution status, robot's tasks, warnings, and errors that may occur during assembly. Especially, the operator may be equipped with multiple smart wearable devices, such as Augmented Reality (AR) glasses, smartwatches, or smartbands and tablets according to the use case specifications. For that reason, developments regarding wearable devices are mandatory for direct communication between the operator and the system.

In Task 5.3, the main outcome will be a real-time digital twin for process reconfiguration by the combination of received data from various sources. A Digital Twin (DT) is considered as a bridge between the real and digital world. To effectively represent the whole workstation area, this module must interact directly with the real world. More specifically, the digital twin ingests information about the workstation area layout, the resources (robot operations, robot state), and the different parts (consumables, assembly parts, fixtures) that exist in the real world. The proposed DT is hardware agnostic and could integrate seamlessly multiple robots and sensors. Real-time control and monitor of the workstation are achieved with the use of different sensors.

An important outcome that is foreseen to be implemented in WP5 is the creation of a fenceless human-robot collaborative assembly. Especially in Task 5.4, the goal is for the robot to be able to adapt to changes regarding the process and human intention by creating collision-free trajectories. The two major components that will be developed in this task are the Human Body Detection (HBD) and Human Task Prediction (HTP). These are capturing the data from different types of sensors, such as 3D vision sensors, 2D laser scanners, and wearable devices as mentioned previously.

Finally, the last expected outcome of WP5 is the dynamic process execution and quality control. In Task 5.5 an AI controller will be developed to control the production resources and coordinate them. The Execution Control is the responsible component for controlling the overall process execution, able to handle all the involved resources. It can adjust the production flow based on any changes that may occur according to the information that is captured in the DT. More specifically, this module will enable the flexible reconfiguration of behaviors between resources, to provide a safe and collaborative setup where the operators and robots can work together.

1.2 Structure of the deliverable

This deliverable is structured into 6 sections including the introduction. The work package is divided into five tasks. The first task is focusing on the requirements and specification gathering for the foreseen development. The work of this task is documented in this report. The other tasks are linked with specified components that will be developed during the project. Section 2 presents the state of the art in the following major components: execution control, digital twin, human detection, task prediction, and user interfaces. Section 3 gives an overall presentation of the architecture of WP5 and how the components will be integrated into a common system. Section 4 describes further each component followed by section 5 which presents the captured requirements. Finally, in section 6 are presented the two use cases where WP5 developments will be deployed and tested.

2. State of The Art in real-time control and actuation

2.1 Execution control and reconfiguration

In common manufacturing cases the control of the system is limited to the control of a production cell. The current state of automatization in manufacturing is based on Programmable Logic Controllers (PLCs) which are microprocessor-based digital computers. Although this technology first appeared in the late 1960s and is characterized as deterministic, it is still an active field of research. Tools that have been developed in PLCs' field mainly focus on the detection of faults. In [1], an automated tool named Manufacturing Process Failure Diagnosis Tool (MPFDT) is presented, which can detect and isolate faults in a PLC-controlled manufacturing system. One more tool called PLC Log-Data Analysis Tool (PLAT) was proposed [2] for identifying operational faults and behavioral anomalies that may occur during the execution of the production. Figure 2 provides a traditional automation pyramid.

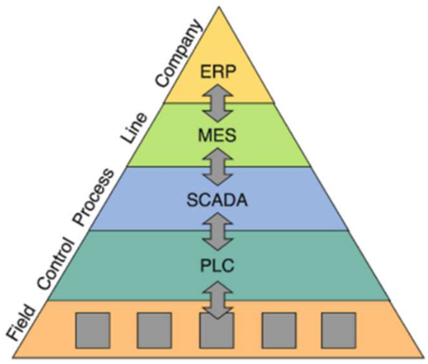

Figure 2: Traditional automation pyramid [3]

In past years, the concept of dynamic reconfiguration is proposed by the academic literature as a solution towards Industry4.0 and flexible manufacturing systems [4][5]. The reconfiguration trend has taking considerably importance in both, the academy, and the industry, given the trends of volatile demands and emerging markets. This trend has stablished several needs, such as rapid modification of processes, reconfigurable hardware and software, and productions scalability, impacting directly legacy components such as PLCs controllers. To cope with the strictly deterministic character of PLCs more flexible methods such as Finite State Machines (FSM), and Behavior Trees are proposed. Software modules for coordination of the execution based on FSMs have facilitated the incorporation of high-level behavior within robots and PLC-based systems by providing a suitable interface with runtime modification capabilities [6]. FlexBe is an example of such a software used in many robotics applications [7][8][9].

Behavior Trees (BTs) are mathematical planning models that are suitable for solving complex applications [10][11]. Initially, they were developed for the computer gaming industry to increase modularity in control structures [12], but BTs are suitable for a wide range of applications such as robotics, computer science, and control systems. In [13], a unified framework of behavior trees is represented, alongside the theoretical and mathematical background. More specifically in robotics applications, there is a wide range of applications. To begin with, in [14] two algorithms of BTs were developed, the first is based on Atomic BTs for each postcondition and the second one is based on iteratively expanding a BT from the Goal Condition. An end-to-end system named CoSTAR was developed that has the capability of performing different tasks that should be robust to environmental changes.

In current market situations, quality is considered part of the requirements to stay ahead of competition. Even though some processes work with relatively low defect percentages, the severity of their implications can be considerable depending on the risk involved in using badly produced goods. Models are essential for the deployment of the control processes and, at the same time, facilitate revealing hidden flaws (e.g., product and processes deterioration conditions) that couple the status of the production process with operational conditions. Diagnostics involve big data sources and their analysis through various AI components. It performs detection and diagnosis of abnormal conditions of processes and, thus, performs a causal analysis. Prediction and Data-driven forecasting is widely used in different processes, including predictive maintenance and quality prognosis [15], which helps in the improvement of quality and safety of the manufacturing processes.

Based on the state of the art that was analyzed before, in the context of the dynamic reconfiguration advanced ways for controlling a manufacturing system with multiple resources are required to be developed. More specifically, following the evolution of technology, Behavior Trees in combination with AI modules seem to be the most suitable methods to be used in a non-deterministic environment.

2.2 Digital Twin

The integration of Digital Twins is a compulsory component in smart manufacturing systems [16]. Several opportunities for using digital twins to address the complexity of collaborative production systems are explored in [17]. Gkournelos et al. [18] proposed a scalable assembly execution control framework based on real-time process reconfiguration and robot behavior adaptation by developing object detection, global navigation, and accurate localization modules. Kousi et al. [19] investigates the use of Digital Twins to enable system reconfiguration. A simulation platform for human-robot interaction was developed that enables robots to avoid obstacles or humans that appear in a workspace [20]. A novel process evaluation method was proposed in [21]. It embeds three methods: a real-time mapping mechanism, construction of the twin-based machining process evaluation framework, and process evaluation driven by digital twin data. In [22] researchers provide an object-oriented event-driven simulation as a digital twin in a human-robot collaborative cell that combines different methods of manual assembly in HRC. All the related work around the DT implementations is focusing on use cases aiming to improve one specific operation and not giving focus on the data modeling and management.

Development of WP5 Digital Twin requires the use of multiple technologies mentioned above. Complex and dynamic environments such as a human-robot collaborative assembly system arise the need for better data management. Taking into consideration this need the creation of a Digital Twin for execution reconfiguration, real-time process reconfiguration and robot behavior adaptation are two mandatory components that have to be implemented in the context of ASSISTANT.

2.3 Human detection and activity tracking

The human behavior is the main unpredictable factor in such a collaborative environment. It can cause changes or issues in both production execution and safety. To ensure the prevention of issues in the production line, human body detection and human task prediction are crucial.

In the last years, deep learning techniques have greatly increased accuracy of the detection of human figures. Human skeletons pose and hand fingers can now be detected with a simple RGB camera, which previously was only possible in the context of animation studios with special equipment. Driven by recent computer vision and robotic applications, determining 3D human poses has become increasingly important and attracted growing interest. Furthermore, the major factor that is mainly unpredictable in such a collaborative environment, is human behavior, which can cause changes or issues in both production execution and safety. To ensure the prevention of issues in the production line both human detection and task prevision are crucial.

There is extended research on human modeling and monitoring [23]. Andrianakos et al. [24] proposed a solution for monitoring the execution of human tasks. This solution is based on object detection and hand detection using in parallel machine learning techniques. Another way to achieve human modeling was proposed by M. Urgo et al. [25] combining a human pose estimation with a statistical model for an operator's task identification. In [26] researchers suggested a solution based on the Dynamic Time Wrapping (DTW) algorithm, which is based on the measurement of similarity between two temporal sequences. In [27], we see a solution approach similar to the one in [25]. In this work a product assembly task is modeled as a sequence of human motions and Hidden Markov Models (HMM) are used for solving the human motion prediction.

Driven by recent computer vision and robotic applications, recovering 3D human poses has become increasingly important and attracted growing interest. Skeleton data can be extracted using open-source software, OpenPose [28]. Another solution with a simple yet effective self-supervised correction mechanism was proposed in [29]. On [30], researchers aimed to minimize the error in mapping from a limited 2d pose into 3d positions. A similar solution to the problem of projecting 2D joints into a full 3D mesh was tested in [31] This is a challenging problem due to the complexity of the human body, articulation, occlusion, clothing, lighting, and the inherent ambiguity in inferring 3D from 2D.

The investigation around these technologies is driven by the need of increasing the efficiency of HRC. ASSISTANT solution aims to provide a safe and seamless collaboration between the involved resources by exploiting the recent technological advances in AI.

2.4 Human Side Interfaces

Human-Robot Interaction (HRI) plays a crucial role in the industry. One of the main challenges in new human-robot collaborative systems is the communication between robots and operators. The oldest and the most challenging human-robot Interaction is the physical Human-Robot Interaction (pHRI) [32] which is based on teleoperation principles. Especially, pHRI uses Wired Control Interfaces and various 6D devices (joysticks, small manipulanda, etc.).

In recent years HRI is supported by wearable and smart devices, such as smartwatches [33], smartbands, AR glasses, and several more. Those devices can either receive or provide much information to the human that is crucial for the assembly. Villani et al. [34] propose an advanced troubleshooting application that enables the interaction with smartwatches to assist workers. In [35], a smartwatch application enhances the synergy between humans and robots in a collaborative assembly cell. Alongside the smartwatch application, an AR application was developed to control the overall task execution and other useful functionality. Augmented Reality (AR) is an interactive technology that has gained interest during the last years. Researchers in [36] developed an AR tool that supports the HRI, more specifically the interaction with a robotic arm. A more complicated AR application was developed in [37]. It enables quite direct interaction between humans and robots. More specifically, it allows the operator to give direct instructions to a mobile platform and receive information about the robot's state through an AR headset.

Taking into consideration the current state of the technology in terms of human side interfaces, in ASSISTANT project we aim to develop applications which, as mentioned above, enable the synergy between humans and robots. This should be achieved in an easy and intuitive way. Non-expert users should be able to interact with the system.

3. Overall Architecture

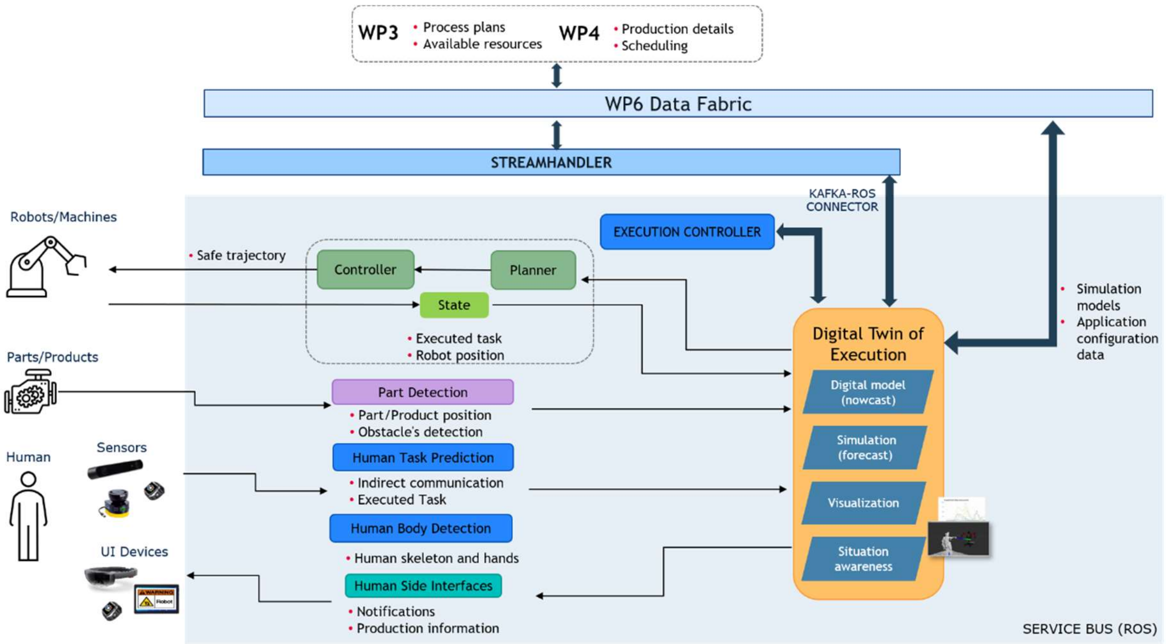

Figure 3: WP5 architecture diagram

WP5 is responsible for creating a communication bridge between robots or machines and operators. Multiple objectives related to real-time actuation and the placement of humans at the core of technology relate to this WP. The Digital Twin of Execution (DTE) provides lots of information about the current state of the production. The operator will be informed about errors that may occur, the current state of the production, and also the next steps for accomplishing a specific task. This is achieved using UI devices such as AR glasses, smartwatches, and tablets. Furthermore, WP5 is responsible for developing flexible cognition methods and resource handling for collision-free path planning and adaptive motion planning in parallel with the estimation of human intention and safe trajectories on the shop floor.

Figure 3 shows the interaction between all the components that will be developed in WP5. To begin with, HBD and HTP are the main two modules that give input to the DTE regarding the execution status of tasks and human status. In parallel with the operator's information, the DTE must be aware of much information regarding robots. For that reason, bidirectional communication is created between the DTE, and robots/machines based on the interface of each robot constructor. By combining those inputs, the DTE in turn, informs the Execution Controller that dispatches to the robots or operators their next task.

To enable seamless integration and facilitate data communication among the technical tools developed in ASSISTANT, the project develops the Data Fabric - a system abstracting data storage and management and providing a unified interface for data access and communication. The WP5 digital twin integrates with the data fabric both directly (e.g., by storing simulation models or application configuration data in data fabric services via APIs and networked service interfaces) and indirectly (e.g., when instrumentation and monitoring data is streamed from production environment equipment to the StreamHandler platform, which after processing makes use of the data fabric services for data storage and archival). In addition to fundamental data storage and management capabilities, the Data Fabric also provides capabilities for deep integration with tools through the exposure of domain models. These domain models capture knowledge and intrinsic details of the operational context of tools (e.g., configuration data and simulation models for the digital twins developed in WP3-5) in machine readable format but do so using language constructs and data structures designed for the application domain (as opposed to the generic storage services of the data fabric that are agnostic of the operational context of the tools storing the data). The ASSISTANT data fabric is designed as a layered architecture with distributed services interconnected using networked interfaces.

The WP5 digital twins (like the other tools developed in the project) integrate with the data fabric through APIs exposing links to both the data fabric services, as well as the domain models developed specifically for the WP5 digital twin. To enable integration with the ASSISTANT data fabric, data fabric clients (such as the WP5 digital twin) must be capable to render machine-readable versions of its operational context (e.g., the state of simulations or parameters of configuration) in established formats (e.g., JSON), as well as have the network connection capabilities needed to make use of the networked APIs and interfaces the data fabric exposes for, e.g., storage or search of this data in the fabric) and the StreamHandler platform (for time series-based monitoring data). As the data fabric is a unified platform for all digital twins and tools developed in ASSISTANT, the requirements for integration and use of the data fabric services are documented in deliverables from the responsible work package (WP6). For more details on the requirements for integration with the data fabric, and the architecture of the ASSISTANT data fabric, see deliverables D6.1 and D6.2, respectively.

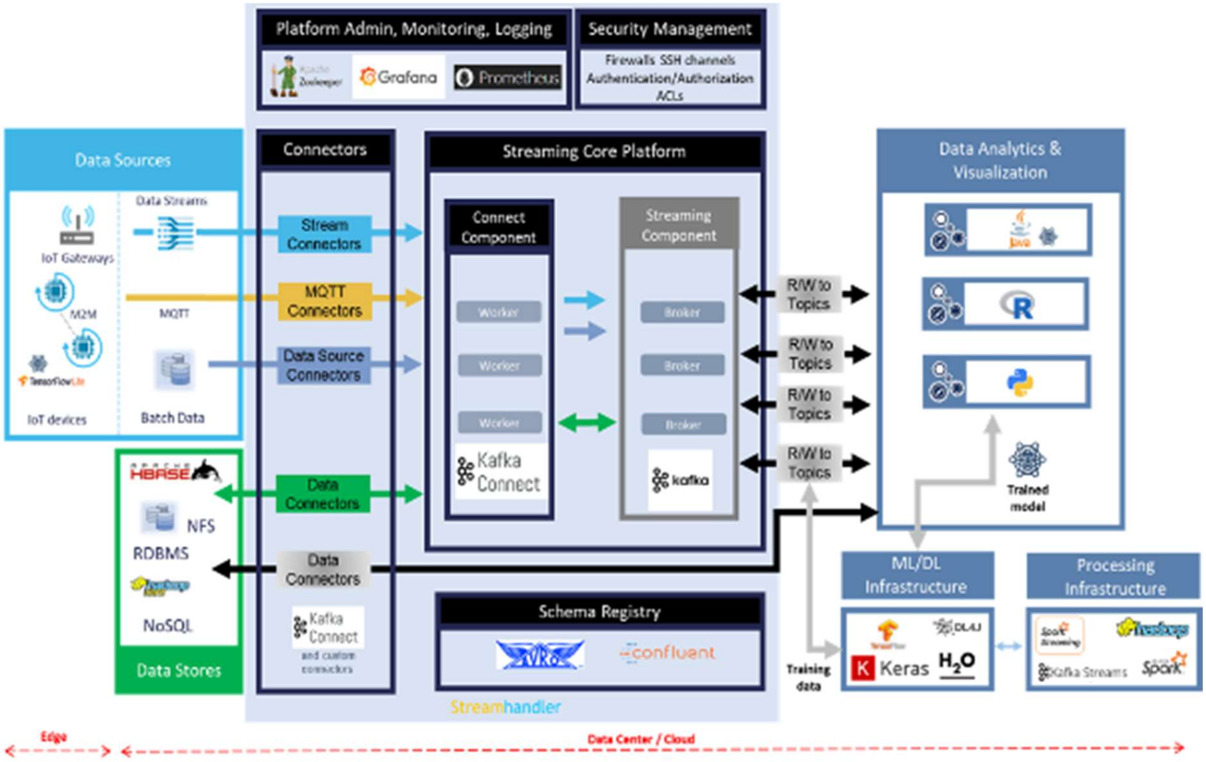

Figure 4 StreamHandler Platform - High-Level Architecture

In the ASSISTANT project, StreamHandler will be used as the component collecting run-time data derived from the shopfloor and storing them in the Data Fabric component. For this functionality, specific connectors will be utilized to transfer the data from the shopfloor to the cloud and from the cloud to the Data Fabric infrastructure. Apart from the Data Fabric infrastructure, StreamHandler will be used for run-time applications like monitoring and dashboards. It is a high-performance distributed streaming platform for handling real-time data based on Apache Kafka with demonstrated low latency and high throughput. It can efficiently ingest and handle massive amounts of data into processing pipelines, for both real-time and batch processing. The platform and its underlying technologies can support any type of data-intensive ICT (Information and Communication Technology) services (Artificial Intelligence, Business Intelligence, etc.) from cloud to edge.

The key capabilities and features offered by the platform are run-time monitoring and event-processing, interoperability with all modern data storage technologies and popular data sources, distributed messaging system, high fault-tolerance, elasticity - high scalability, and security. As depicted in Figure 4 StreamHandler platform consists of Integrated Connectors, Streaming Core Platform, Schema Registry, Security Management and Platform Admin and Monitoring Dashboard.

The analysis of the product's state (and its predictions) will allow real-time reconfiguration of the production plans in case of deviation of the planned process and, therefore, rapid reconfiguration of the production process depending on the type of deviation. In other words, the deviation can lead to reprocessing of the current product, or its elimination followed by a new product production.

4. Component Description

The whole WP5 implementation consists of five basic components to be developed. The overall framework of WP5 can be considered as a control loop. First, execution control and reconfiguration are responsible for informing both robots and humans of the execution process dynamically. Robot execution status can be easily obtained since it is programmed. The major uncertainty factor in production is human behavior. For that reason, the identification of tasks and their execution status as well as the operator's position cannot be observed easily, and for that reason, two main modules will be developed: Human Task Prediction and Human Body Detection. Both results gathered from the robot's and operator's state, are fed into a Digital Twin which is responsible for informing the assembly execution controller.

4.1 Execution Control and Reconfiguration

Execution Control and Reconfiguration is responsible for adjusting the production process based on the data gathered from the Digital Twin, as previously defined. It is considered as the general controller that can control the whole production in real-time. The basic idea is to use the combination of information gathered from WP3 and WP4 to specify the desirable industrial conditions to be performed in the production process. WP3 is responsible for providing the process plan, while WP4 is responsible for providing production details (e.g., production plan decisions and scheduling decisions). The output of these WPs is transferred via the Data Fabric (i.e., WP6, responsible for storing multiple information in Data Fabric) into the control process. This information will be used by WP5 that will help to settle the real-time adaptation of the system and the execution, production, and evaluation of process quality control.

Regarding quality control, the combination of historical and streamed statistical analysis and machine learning methods will use the information to evaluate and predict both the state of the process (machinery and equipment conditions) and the state of the products. Based on the information previously stated, in ASSISTANT the focus of process and product quality control will be principally on predictive maintenance through Fault Diagnosis and Fault Detection (FD) or Remaining Useful Life Prediction (RULP), and quality inspection. Even though there are considerable challenges involved in the product and process quality control, an incremental approach will be performed. This implies that first standard methods and techniques will be developed for incorporation of these techniques in the ASSISTANT framework to, later, extend the capabilities of the developed components depending on achievements obtained on previous stages. It must be highlighted that an initial focus on using human-centric and ethical frameworks will be developed, and therefore, the initial focus on development will be dedicated to the black box model challenge (i.e., explainability). This implies that technical and non-technical approaches will be studied with a perspective of a general manufacturing framework.

The main component of the Execution Controller will be the Process Orchestrator. This component will serve as the entry point of the WP5 developments since it will be responsible for triggering the cell's digital twin to perform the related process. It will also be indirectly connected to the task scheduler/planner (WP4) by retrieving the product/process/resource assignment and the Process Planner (WP3) by retrieving the production process to be executed in the production line. The Quality control module will monitor production and provide feedback to the Process Orchestrator regarding the process in a closed-loop manner. Once a defect from the Quality Control module is identified or a problem during a process execution occurs, the Execution Controller in an online fashion will communicate the error to the Process Planner for evaluating and providing alternative process plans as well as to the Production Scheduler for adapting the schedule accordingly (i.e., continue with other products and/or resources).

Figure 5 below shows this communication workflow between these modules. It shows data retrieval (orange arrows), cooperation/execution (green arrow), and error handling (red arrow). The communication will be achieved through the development of specific services for handling these exception cases but also through the Data Fabric by retrieving information generated by Production Scheduler/Process Planner.

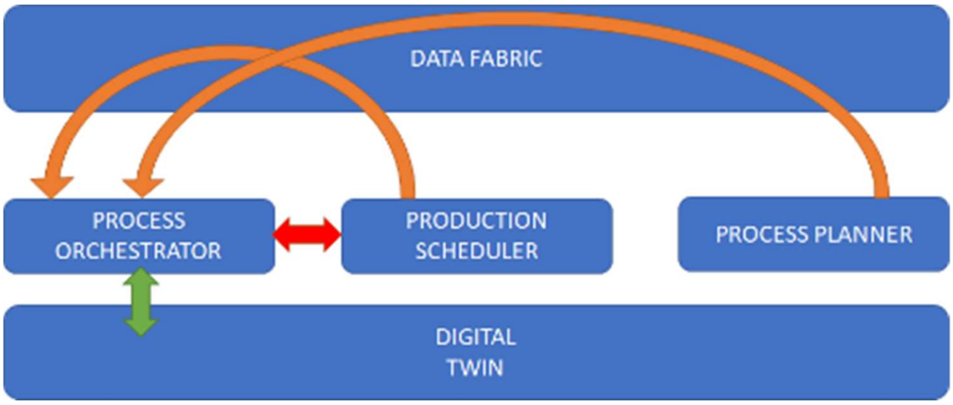

Figure 5 Process Orchestrator communication

4.2 Digital Twin of Execution

To effectively represent the whole workstation, the Digital Twin of the Execution (DTE) module must interact directly with the real world. More specifically, the DTE consumes information about the workstation area layout, the resources (robot operations, robot state), and the different parts (consumables, assembly parts, fixtures) that exist in the real world. The proposed DTE is hardware agnostic and could integrate seamlessly multiple robots and sensors.

The precise simulation that comes from the DTE, integrates domain modeling and simulation with accurate data. Domain modeling is to model a world consisting of machines, operators, and environmental states. This is important for the design of control algorithms. In the AI design lifecycle shown in Figure 6, there will be first an offline effort of analysis, to discover correlations and to analyze them in detail. Once this insight is gathered, an AI model can be created, trained and pre-validated on historic data. Once this model is deployed and executed, it interacts with the machine, operator, and environment by reacting to sensor input. Constant monitoring and validation of the runtime behavior builds confidence in the model or allows the data scientist to adapt the system (either the model or the operations) to achieve better performance.

The added value of the domain modeling effort is obvious in both offline and online. In the offline part, the domain model allows for structured storage and access to the historic data. In the online part, the domain model represents the world model with which the real-time functions interact (such as the runtime model). The online domain model is usually quite tieddown to its online purpose and is less generic than the domain model in the offline world. For performance reasons, the domain model might even not be explicitly visible anymore in the online implementation, although implicitly it is still at the core of how the online data is structured and processed.

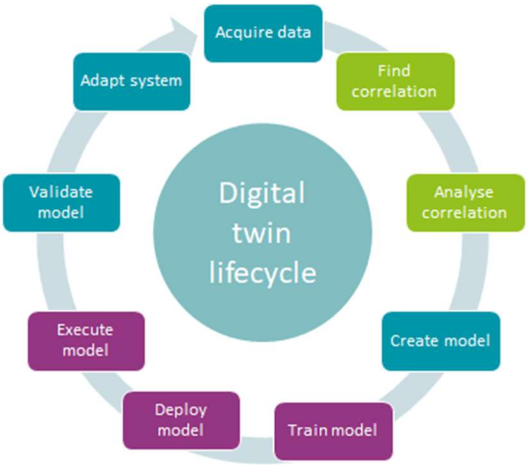

Figure 6: Digital Twin lifecycle diagram

The real-time awareness of the workstation is achieved with the use of different sensors. Sensor data needs to be captured and fitted to provide information for keeping up to date with the digital models. More specifically, the sensors' combination provides the input to components for detecting the position of humans or robots and the execution status of their tasks. Over and above the information for the physical world, the DTE must be aware of production details. Data Fabric is responsible for providing those data into a data model. A critical aspect that the DTE must be aware of is the Bill Of Processes (BOP) that contains the process level description of the production system. In addition, various information regarding the whole production is crucial. At this point, it is worth mentioning that this data needs to be anonymized to prevent infringement of rights. Apart from retrieving data, the DTE provides feedback to the operators through the smart interfaces that are connected with the system. The operator is informed about the execution status, robot's tasks, warnings, and errors that may occur in the assembly.

A hierarchical model of the data that are handled within WP5 is presented in the following diagram (Figure 7). The used color-coding maps the common classes of the data structures. This is an initial design of the data model and will be updated through the course of the project. Data handling functions will be developed for exploiting the Data Fabric interfaces and updating this model. This will maintain uniformity across the various data types utilized in each ASSISTANT work package.

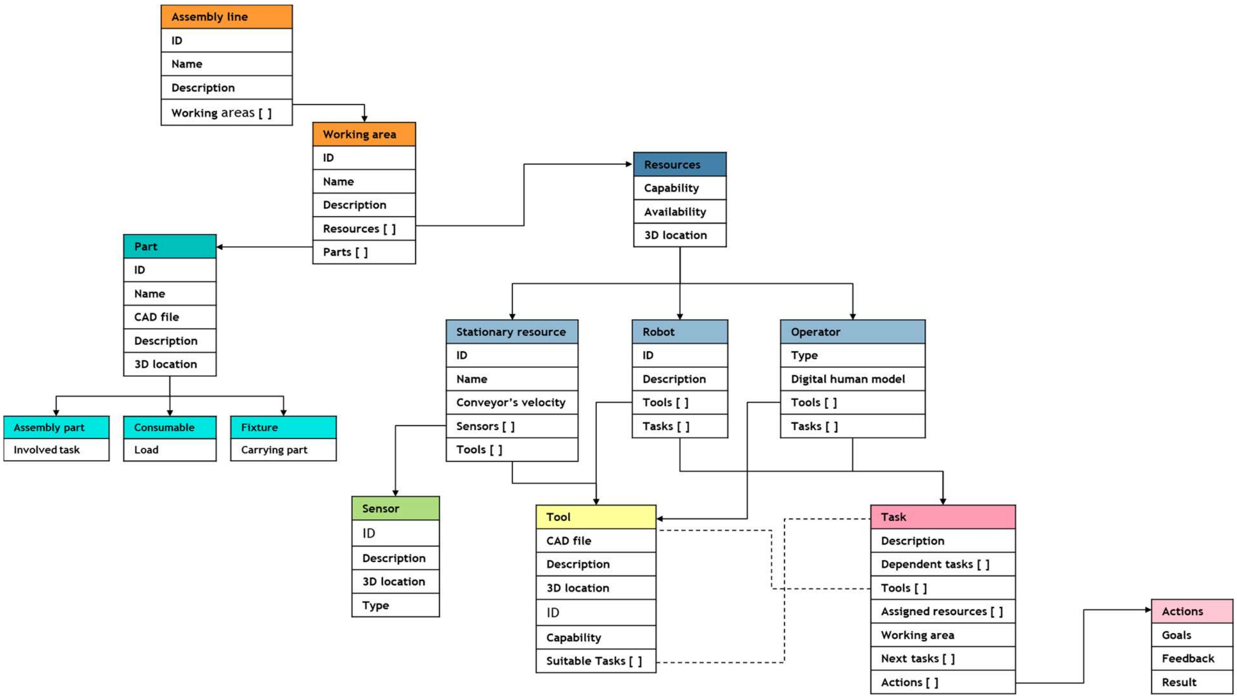

Figure 7: Representation of Digital Twin Data Model

In Figure 8 a prototype of the simulation environment is presented on the left and the digital twin representation on the right. The simulation is built on GAZEBO physics simulation [38] capable of simulating many robots and generating emulated sensor data. This method facilitates the deployment and testing of WP5 components prior to physical installation. The consolidated information that is available through the DTE can be represented in the 3D space. This DTE focuses on the modeling of human and robot behavior as long as the execution status. More information about the human side can be found in the next section.

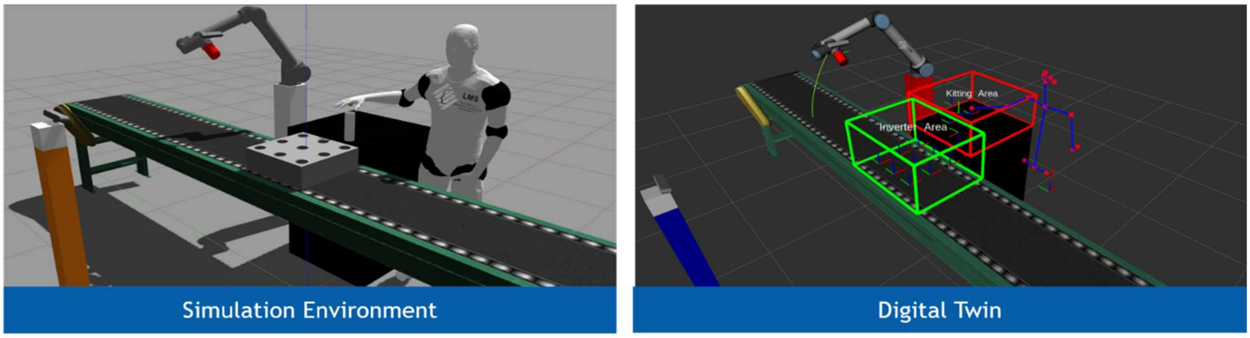

Figure 8 Simulation Environment and Digital Twin representation

For modeling the robot's behavior, we have identified three required components: the State, Planner, and Controller. Robot state retrieves data from the robot and presents the joint configuration, the IO states, and the tool position/orientation. The Planner is responsible for accepting position goals and calculating a collision-free trajectory for these goals. The Planner uses the planning scene to check for collisions online. The planning scene is part of the DTE and consists of static objects like fixtures, and tables and dynamic obstacles like the human operators who are moving. Finally, the Controller moves the robot along the given trajectory.

This component will work with various robot brands. The modeling of the overall execution and the tasks that are assigned to the resources is the responsibility of the Execution Controller. As mentioned in Section 4.1, the DTE will be connected with the Execution Controller for exchanging data on the status of the production.

4.3 Human Body Detection and Human Task Prediction

Aiming beyond the standard means of safety detection that causes emergency stops on robots when the worker is near, advanced information is required in parallel with direct interface with robot trajectory execution. In the actual assembly line, operators and robots follow a predefined order of tasks. While the tasks that are executed by robots can be monitored, human processes cannot be fully tracked. Operators have several degrees of freedom, which means that the way a task is executed defers from person to person. In addition to this, an operator can make additional moves such as touch his head or adjust his uniform. This is a deviation from the predefined order of actions which should not be considered as an error. Human Body Detection and Human Task Prediction are components that need to be developed to create a more efficient production. Human Body Detection can be considered as an input of the Human Task Prediction because it provides useful online 3D skeleton data and basic primitive behaviors. Human Task Prediction aims to predict human intention and task status taking into consideration the different ways a task can be executed and the minor deviations that may occur during execution.

4.3.1 Human Body Detection

The detection of human presence is a mandatory component in an H-R collaborative assembly system for ensuring the safety of the operator. The robot follows a predefined trajectory to execute its task. By detecting the accurate position of the operator, it must be ready to adjust the trajectory to avoid the operator and ensure human safety. Beyond the safety of the operator, replanning of the robot's trajectory can lead to more efficient production, since the emergency stoppage time is eliminated. Human Body Detection (HBD) provides more detailed information regarding the human body's posture and position in the workspace. 2D and 3D data are fused for providing the position of the whole human body. Figure 9 provides the integration of HBD with the robot planner.

Figure 9: Integration of HBD with the Robot Planner

The implementation of HBD is based on ROS principles and requires lots of data that are gathered from vision sensors. The setup of the collaborative workplace consists of stationary 3D cameras placed around the operator, to have good visual coverage. These 3D sensors import information to the Human tracking algorithm. A real-time software quite accurately detects multi-person human body-parts' key points such as hands, legs, core, and head. This algorithm uses an AI model with an internal representation of the human skeleton to avoid problems with occlusions that other approaches suffer from, for example a synchronization of 2D body parts detection with a Pointcloud. By joining those points, the visualization of a human skeleton and its position can be extracted. A Kalman filter is also used on the detected 3D body parts to smooth out sudden movements and noise and provide a more accurate representation. The tool responsible for visualizing the above-mentioned information updates the Digital Twin. The skeleton data are considered as Collision Objects, so the next step is the Pose Update of the extracted key points. The planning scene is updated with the human skeleton points and the Robot's collision-free path is created.

4.3.2 Human Task Prediction

The Human Task Prediction (HTP) component is used to extract important information for improving the worker's collaboration with the system. Such information could be that the operator completes a task so the robot could continue the production procedure, or that the operator needs the assistance of a robot for completing a task. This "communication" between human and robot is currently established through buttons that directly inform the system about the worker's status. HTP provides indirect communication, and it is not imposing on the operators a strict workflow. The HTP is built up of probabilistic models that can cope with the errors and noise in measurements. Two techniques are examined for the design of HTP and will be validated during the project. The first one is a stochastic approach based on the statistical modeling of Hidden Markov Models (HMM). The second method is based on the AI Deep Learning approach.

The initial setup of the collaborative workplace consists of a stationary 3D camera placed in front of the operator as shown in Figure 8, to have good visual coverage of the human, IMU sensors attached on the operator's hands, and an AR headset. The HTP module uses also semantic information from the workplace. The workstation is divided into several schematic areas which are labeled according to heuristic knowledge on the use case. Examples of such areas could be "kitting" areas, "assembly table" "machines' fixtures" etc.

As mentioned above, the first approach is based on HMM. The combination of the above-mentioned extracted information is fed into a probabilistic Hidden Markov Model (HMM) that achieves human task monitoring. The use of this model requires state modeling. The observation sequence is declared as O = {O1, O2, ..., Om}, where "m" is the number of discrete observations, while "s" and "t" are the initial and the final observations, respectively. An example of this modeling for a "Pick-and-Place" task is presented in Table 1.

Table 1: Description and Requirements of HMM

Observation	Description	Requirements
s	Initial state	• Cylinder detected in kitting area. • Hand away from object
O1	Grab cylinder	• Cylinder detected in kitting area. • Hand wrist detected in kitting area. • Hand wrist detected really close to cylinder.
O2	Pick cylinder	• Hand wrist detected close to cylinder. • Hand and cylinder outside from kitting area
O3	Place cylinder	• Cylinder detected in inverter area. • Hand wrist away from cylinder.
t	End of "pick and place"	• Cylinder in inverter area • Hand wrist outside of inverter area

The second approach that will be analyzed is based on deep learning methods. The AR headset shares the same field of view with the operator eliminating the problems of occlusions. From this perspective and in combination with the IMU data from the smartbands, we can monitor the movements of the operator's hands and extract the key points of those. With that information, we can classify predefined gestures like grasping or releasing an object. This classifier will be given as input to a trained Long short-term memory (LSTM) network that will predict the activity of the worker. For the training process, a testbed of the industrial use case will be deployed on laboratory environment. Specific tasks performed by researchers will be recorded. The trained LSTM will be tested with the overall system. In the next phase after the initial validation, we plan to transfer the testbed to the industrial partners' premises to capture real workers' behavior.

4.4 Smart Human Interfaces

Smart Human Interfaces are the key developments of task 5.2 Worker in the loop. This task is responsible for developing the human-side interfaces that will ensure the human-centric scope of the system. Humans are the most important aspect in production decision-making, and with the right tools, efficient cooperation with the rest of the resources is enabled. Considering the manufacturing system control, ASSISTANT identifies two critical roles that should be able to interact with the system: **Production Managers** and **Operators**. Dedicated interfaces on smart devices will be developed for the Production Manager to check the execution status as coordinated by the AI-based controller and to evaluate possible recovery plans generated by the controller. For the Operators, the given tools will suggest operations and setups to perform, as well as process instructions for assisting their work.

4.4.1 Operator Support Application

During a shift, the operator needs to be informed about several issues such as the execution status, task sequence, warnings, and errors that may occur in the assembly. For that reason, an Operator Support Application will be developed that aims to assist the worker by providing information about the current state of the production. Taking into consideration the work environment and the efficiency, the aforementioned application will be deployed as an AR application. The application’s main goal is to fully support the operator during the execution of the tasks.

The provided AR application aims at supporting the operator during his shift inside an unpredictable and complex workplace. Since this application combines both the virtual and the physical world, a connection between them is necessary. Once the server is connected, the user has to look around and locate the reference object in order to achieve the application’s calibration. From the moment the application is calibrated, as shown in Figure 10, it is responsible for providing many pieces of information that increase the effectiveness of the production through the following functionalities:

- Operator’s task information
- Production information (human information, available resources, and production schedule)
- Early warnings and Errors that may occur in the production
- Information about resources

Access to those functionalities can be achieved by voice commands, air taps, custom hand gestures as depicted in Figure 10. After the initialization of the application, the operator will be able to check the Robot Status that will contain information about the situation of the robot (working, warnings, etc.). Anytime the operator says “Menu”, he/she will be able to see a Main Menu panel that contains Network connection, Workplace calibration, Operator’s tasks, and Production info (Human information, Available resources and production schedule). Operator tasks can be accessed by custom hand gesture. More specifically, the moment the operator makes a special gesture, a pop-up panel will appear, in which operator’s current task as well as a brief description of it will be written. Similar to this, the direct look at the robot will appear a pop-up panel containing robot’s current task as well as a brief description of it. Finally, as mentioned before, the application also supplies the operator with data that can ensure his/her safety. Warnings and errors may occur in the production as well as Robot Trajectories can appear in his POV whenever it is needed.

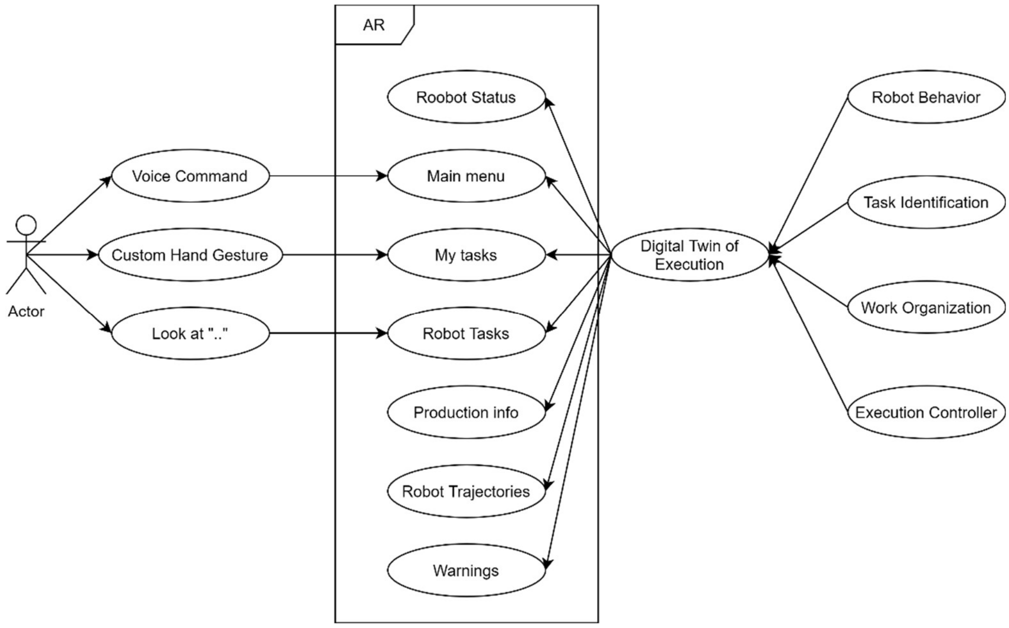

Figure 10: Operator’s AR Application Use Case diagram

DTE is the responsible component to gather needed information for the implementation of the application. As depicted in Figure 10, Robot Behavior, Task Identification, Work Organization and Execution Controller are the modules that parse the needed data as input to it.

4.4.2 Production Manager’s Application

Apart from the Operator Support Application, a Production Manager Application is necessary to be developed. In comparison with operator’s AR application, production manager’s application consists of different features that focus more on the overall production management. It is crucial to be able to check the execution status and evaluate possible recovery plans. This application will be served as a web app running on every device independently of its operating system (Windows, Linux, Android, etc.).

As shown in Figure 11, Work Organization, Execution Control, and Production Planning are responsible for providing information as input to the Digital Twin, which provides feedback to the production manager through the Web framework. The manager can receive notifications in his/her tablet about the whole production's execution status or the generated task allocation plans to evaluate possible recovery plans in case something goes wrong in the production. Furthermore, the manager will be able to view the different KPIs that are currently achieved with the running execution schedule. In parallel with the abovementioned functionalities, Production Manager will also be able to choose alternative plans.

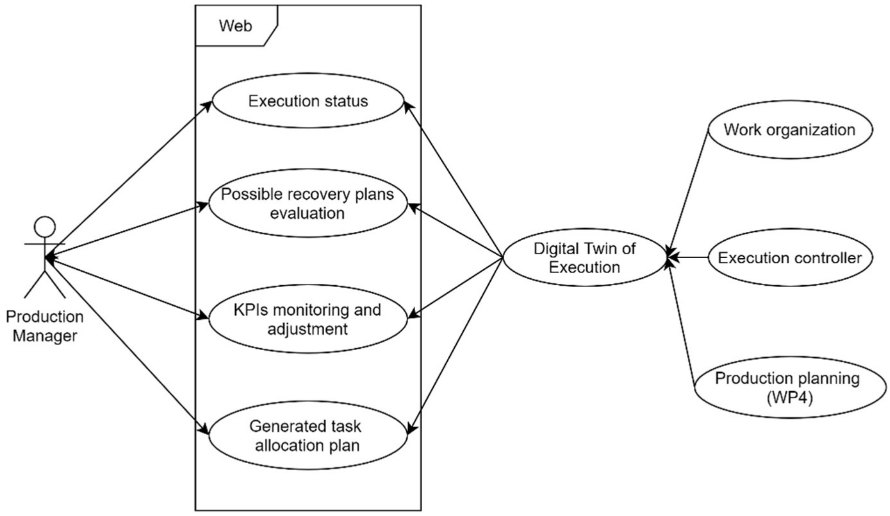

Figure 11: Production Manager's Application Use Case diagram

5. Requirements Definition

Together with the definition of the individual modules that WP5 consists of, this section provides requirements that are connected with its implementation. Those resulted from different ways of analysis such as brainstorming, interviews with end users, review meetings, and document analysis. First of all, the requirements were categorized by development, then classified by category (Base, Reasoning, Interface, Processing, Development, Analysis, Communication, Computation, Display, Data storage, Ethical, and Safety) and priority (sorted by the most to least necessary, SS, HP, LP, NH). Finally, their descriptions are presented in the requirements' tables (rationale). To avoid repetition of information, extended information about Requirements Engineering can be found in deliverable D3.1.

5.1 Execution Control and Reconfiguration

Table 2: Requirements of Execution Control and Reconfiguration

R1.1	**Control of resource behavior**	Priority SS	M36
Requirement	This component must be able to control and monitor the individual resources of the production system.		
Category	Base requirement		
Rationale	Control based on resources availability		

R1.2	**Online reconfiguration of the system**	Priority HP	M36
Requirement	This component must adjust the production routing regarding the available alternative process and production plans.		
Category	Reasoning requirement		
Rationale	System adaptability		

R1.3	**Hardware agnostic**	Priority LP	M36
Requirement	This component must work with different types of systems and machines without suffering compatibility issues. An abstract interface must be provided, common for controlling all kinds of resources of a production system.		
Category	Interface requirement		
Rationale	Resource control		

R1.4	**Event and interrupt handling**	Priority DOA	M36
Requirement	It will be capable of handling events that may be cause due to human intervention or unexpected errors.		
Category	Processing requirement		
Rationale	Execution Control and Reconfiguration requirement		

R1.5	**Control and Quality pre-specification of input-output variables**	Priority LP	M36
Requirement	The developer should clearly define input and Output classes/features to be used in the control and quality evaluation processes.		
Category	Development requirement		
Rationale	Execution Control and Reconfiguration requirement		

R1.6	**Correct provisioning of information**	Priority SS	M36
Requirement	Provisioning of information should be sound. Good data is essential to the development of AI components and therefore, depending on the quality and quantity of the data provided some algorithms would not be able to be constructed (e.g., deep neural networks require considerable information depending on the number of features in the analyses).		
Category	Analysis requirement		
Rationale	Execution Control and Reconfiguration requirement		

5.2 Digital Twin for Execution

Table 3: DTE requirements

R2.1	**Real-time data processing**	Priority SS	M36
Requirement	The Digital Twin should process gathered data with a high runtime efficiency to be able to make decisions in almost real-time.		
Category	Communication Requirement, Computation requirement		
Rationale	Data acquisition		

R2.2	**Realistic Representation**	Priority HP	M36
Requirement	The information that will be visualized should be accurate according to the real system.		
Category	Display Requirement		
Rationale	Visualization of the DTE		

R2.3	**Hardware agnostic**	Priority HP	M36
Requirement	DTE must gather data from multiple sensors, tools, machines, and robots. The interface with these elements must be general enough and to not require special adaptation.		
Category	Interface Requirement		
Rationale	Connection with hardware		

R2.4	**Exchange data with Data Fabric**	Priority HP	M36
Requirement	The DTE should communicate with the Data Fabric for retrieving and storing data.		
Category	Interface/Data Storage Requirement		
Rationale	DTE Requirement		

R2.5	**Handle anonymized data**	Priority HP	M36
Requirement	This module should work with anonymized data only. Such data may not link with an individual user.		
Category	Ethical Requirement		
Rationale	Data management		

R2.6	**Accurate simulation of the execution**	Priority HP	M36
Requirement	The resource’s behavior should be accurately simulated under the control of the WP5 components.		
Category	Ethical Requirement		
Rationale	Simulation		

5.3 Human Body Detection and Human Task Prediction

Table 4: HBD and HTP Requirements

R3.1	**Real-time provisioning of sensor data**	Priority SS	M36
Requirement	Sensors should provide data in real time to the HBD and HTP for respecting the maximum response time.		
Category	Communication Requirement		
Rationale	Sensors' data acquisition		

R3.2	**Sensors' accuracy**	Priority SS	M36
Requirement	Data provided by multiple sensors should be fed in the components and consequently in the DTE with a high accuracy. Noise reduction algorithms must be developed to ensure the correct human body detection.		
Category	Computation requirement		
Rationale	Sensor noise		

R3.3	**Anonymization**	Priority HP	M36
Requirement	Ethical reasons do not allow the disclosure of operators' personal information.		
Category	Ethical Requirement		
Rationale	Data management		

R3.4	**Interface with Robot's planner**	Priority SS	M36
Requirement	The HBD's output should be provided to the robot's planner for ensuring that it will execute a collision free trajectory.		
Category	Safety Requirement, Interface requirement		
Rationale	Robot's collision free trajectory		

R3.5	**Use of safety certified functionalities**	Priority SS	M36
Requirement	To comply to the safety regulations of the HRC, safety certified sensors will used, and the robot speed will be limited (DIN ISO 10218-1&2, ISO TS 15066). HBD must not interfere to the use of the safety certified functionalities.		
Category	Safety Requirement		
Rationale	Human's safety		

R3.6	**Simple training of HTP**	Priority LP	M36
Requirement	Training procedure is required for the reasoning of HTP. In order to apply this component quickly on different assembly cases, the training must me easy and quick to perform.		
Category	Analysis requirement		
Rationale	AI learning		

5.4 Human Side Interfaces

Table 5: Human Side Interfaces Requirements

R4.1	**User-friendly interaction**	Priority SS	M36
Requirement	The application should be simple and intuitive to be used by everyone regardless of their expertise and experience.		
Category	User interface requirement		
Rationale	User experience		

R4.2	**Easy configuration**	Priority HP	M36
Requirement	The application should be ready to adapt, e.g. the layout should be easy to modify.		
Category	Functional requirement		
Rational	Deployment		

R4.3	**Avoid overshadowing real world**	Priority HP	M36
Requirement	In an AR system, virtual information should not block real-world information.		
Category	Display requirement		
Rationale	User experience		

R4.4	**Synchronous communication with the DTE**	Priority SS	M36
Requirement	Any change in the actual world must be reflected in an immediate update of the application through communication with the DTE.		
Category	Communication requirement		
Rationale	Back-end connection		

R4.5	**Self-explainable displays**	Priority LP	M36
Requirement	The user of the application must be able in short time to be familiar with the functionalities of the application.		
Category	Display requirement		
Rationale	User experience		

R4.6	**Multiple language support**	Priority LP	M36
Requirement	The applications must display information on the native language of the users.		
Category	Display requirement		
Rationale	Language		

R4.7	**Cross-platform**	Priority LP	M36
Requirement	Applications must operate on various devices.		
Category	Functional requirement		
Rationale	Deployment		

6. WP5 connection with the Industrial use case

WP5 interacts with two out of three defined use cases, STELLANTIS (PSA) use case and Atlas Copco use case. In the following sections we provide a summary of those use cases and their requirements from WP5 perspective.

6.1 STELLANTIS Use Case Requirements

This is the proposed use case for the ASSISTANT project's automotive scenario. The deployment of the STELLANTIS use case is based on a pick and place demo. Robots and operators have been assigned specific tasks to assemble the end product in the same shared workplace. This leads to the need of creating a collaborative cell in which robots can execute tasks with high accuracy and flexibility. Robot must be ready to adapt in changes regarding operator's position or operator's task execution.

WP5 will pursue the requirements related to real-time actuation and putting the human operators at the core of technology. The results of task 5.3 digital twin for monitoring and online reconfiguring of the execution will be demonstrated on this production case. The human side interfaces for Process Engineers and Production Operators will be developed in task 5.2. Finally, the outcomes of tasks 5.4 and 5.5 about a fenceless human robot collaborative assembly workstation will be configured and deployed in this use case. The implementation of the STELLANTIS Use Case is based on specific requirements that are analyzed below.

Table 6: STELLANTIS Use Case requirements

R5.1	**Wait time minimization**	Priority SS	M36
Requirement	The wait time of the robot's should be minimized. This could result in a direct increase in efficiency. When using cobots, human robot coordination may not be totally accurate, and the robot is not running in 100%.		
Category	Productivity		
Rationale	STELLANTIS Use Case Requirement		

R5.2	**Set up time minimization**	Priority SS	M36
Requirement	The time required for a machine or a robot to adjust to changes in the production, new products, or human intention, should be minimized so as not to lose production flow.		
Category	Production reconfiguration		
Rationale	STELLANTIS Use Case Requirement		

R5.3	**Situation awareness**	Priority SS	M36
Requirement	In terms of safety, the operator must be aware of the production's situation all the time. The fulfillment of this requirement is related to robot transparency.		
Category	Human safety		
Rationale	STELLANTIS Use Case Requirement		

6.2 Atlas Copco Use Case Requirements

The Atlas Copco (AC) use case is based on the production of a compressor airend, the core element of a compressor. The compressor airend production is high mix low volume (HMLV) production, and many different variants (up to 30) each consisting of about 20 components) need to be produced.

Figure 12 is a baseline for a full AI lifecycle, including an offline and online part as currently built for the use case at Atlas Copco. It is the result of a workshop held with AC and is representative for an actual deployed AI algorithm in production of the Atlas Copco use cases (adaptive measurement strategy and virtual assistant). The result points out several requirements for the online deployment and execution. During the ASSISTANT project, this simple architecture will be extended to incorporate the complexities of AC's real environment and to validate & verify AC's requirements. Mainly the requirements that follow from the online part are relevant for WP5.

Figure 12 shows the offline and online parts of the setup. Functionally, the offline part uses the historic data in order to train a model that predicts a given production test's outcome. Once this model is able to correctly predict the test's verdict, the model can be used at runtime (in the online part) in order to skip the test and that way save time and money while keeping the quality on par (the adaptive measuring strategy). From an implementation point of view, the offline part is currently deployed with a traditional approach (albeit with a small subset of data to satisfy confidentiality constraints) but will need to be enriched with the insights from WP6 concerning the data fabric. The online part consumes the live data from the machine (again for confidentiality this is currently simulated) and reacts live to the collected data. This part of the architecture is built around a broker (potentially to be replaced with Streamhandler during the project) and some gateways (gwFromOPCUA and gwToDB) in order to process the data. The evaluator service is the one that then executes the functionality foreseen in the use case.

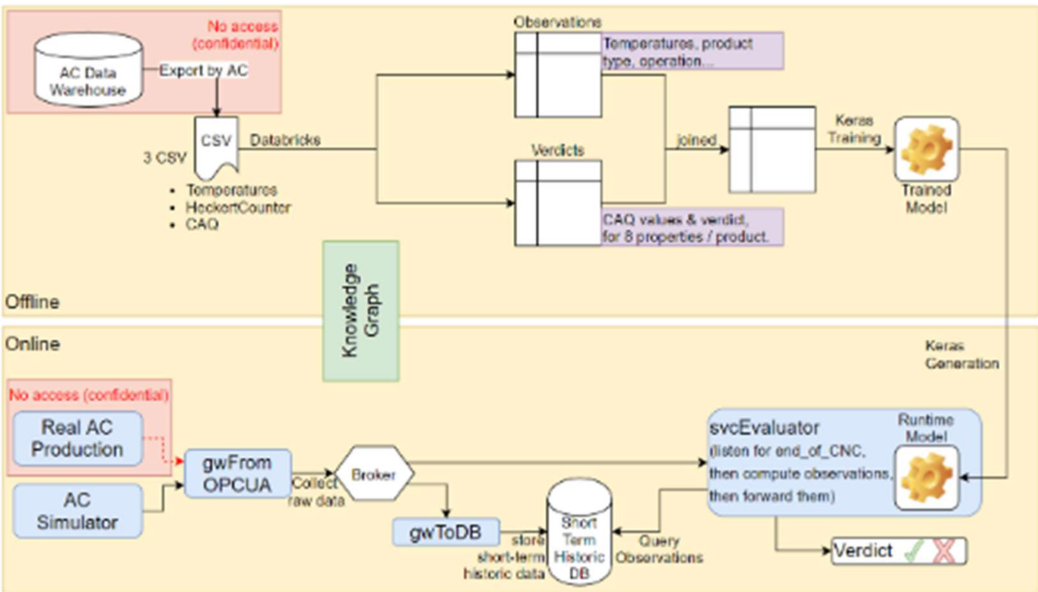

Figure 12: AC case deployment diagram

Overall, WP5 real time control is the package that will deliver the best suited tools to implement our adaptive quality control strategy (adaptive measuring strategy for the CNC machine, CMM machine and manual measurements), and the virtual operator assistant (provide real time suggestions to operators for maximum process stability and efficiency, based on current situation and inputs). We expect the following requirements from WP5 in order to make sure its tools and insights are useful for the realization of the AC use case.

Table 7: AC Use Case requirements

R6.1	**Live data**	Priority SS	M36
Requirement	Connection via OPC-UA to the current sensors should be provided on a near real-time basis allowing for live ingestion of and potential reaction to the sensors' values.		
Category	Connectivity requirement		
Rationale	AC Use Case Requirement		

R6.2	**Historic data**	Priority HP	M36
Requirement	Retrieved data (see R6.1) should be stored in a data warehouse for later (historic) retrieval and processing of the data.		
Category	Historic storage and query requirement		
Rationale	AC Use Case Requirement		

R6.3	**Aggregation of live data**	Priority SS	M36
Requirement	The system should offer computational services for near real-time reactions to the retrieved data (see R6.1). Such computational services should offer aggregations and similar statistical abilities(e.g., average/min/max over a time window). The computational services should also be usable for running the "runtime" part of machine learning models.		
Category	Computational requirements		
Rationale	AC Use Case Requirement		

R6.4	**Traceability of data**	Priority HP	M36
Requirement	In order to allow learning, it should be possible within a production process to trace data from different production steps, data formats, etc. (e.g., by rfid, qr code, ERP system, etc.) so it can be linked to a product, order, machine, operator.		
Category	Traceability requirement		
Rationale	AC Use Case Requirement		

R6.5	**Decrease of design time of AI lifecycle**	Priority HP	M36
Requirement	The time taken to perform an AI lifecycle should be decreased as compared to the current AC approach, by properly leveraging data, information, and knowledge.		
Category	Non-functional performance requirement		
Rationale	AC Use Case Requirement		

7. Conclusion

Real-time control and actuation of manufacturing companies is an area constantly evolving to create more flexible and safe assembly lines. The present deliverable defines key elements and multi-level requirements for reconfigurable manufacturing execution as well as the current state of the technology regarding the individual modules that will be developed. The main purpose of this document is to position the work of WP5 inside ASSISTANT project. In particular, the implementation of the design presented in this deliverable, will be documented in D5.2, D5.3, D5.4 and D5.5. First, the initial design of the real time digital twin will be presented in D5.2 in parallel with the development infrastructure and the design of the interaction with the execution system. Second, in D5.3 the prototype of the digital world model and HRC components will be documented. Subsequently, a prototype of the AI based execution controller, recovery strategies and the interfaces that will keep the human in the loop will be mentioned in D5.4. Finally, the final version of WP5 tools, revised based on the feedback from the use cases validation, will be documented in D5.5.

References

[1] Ghosh, A., Wang, G. N., & Lee, J. (2020). A novel automata and neural network based fault diagnosis system for PLC controlled manufacturing systems. Computers and Industrial Engineering, 139(February 2019), 106188 https://doi.org/10.1016/j.cie.2019.106188

[2] Ghosh, A., Qin, S., Lee, J., & Wang, G. N. (2016). PLAT: An Automated Fault and Behavioural Anomaly Detection Tool for PLC Controlled Manufacturing Systems. Computational Intelligence and Neuroscience, 2016. https://doi.org/10.1155/2016/1652475

[3] E. Y. Nakagawa, P. O. Antonino, F. Schnicke, R. Capilla, T. Kuhn, and P. Liggesmeyer, “Industry 4.0 reference architectures: State of the art and future trends,” Comput. Ind. Eng., vol. 156, p. 107241, Jun. 2021, doi: 10.1016/j.cie.2021.107241.

[4] G. Chryssolouris, “Manufacturing Systems: Theory and Practice”, 2nd Edition, 606p, Springer-Verlag, New York (2006)

[5] Makris, S. (2021). Cooperating Robots for Flexible Manufacturing. Springer Series in Advanced Manufacturing. doi:10.1007/978-3-030-51591-1

[6] McDonald Hayhurst, J. W., & Conner, D. C. (2018). Towards Capability-Based Synthesis of Executable Robot Behaviors. Conference Proceedings - IEEE SOUTHEASTCON, 2018-April, 1–8. https://doi.org/10.1109/SECON.2018.8479047

[7] Conner, D. C., & Willis, J. (2017). Flexible Navigation: Finite state machine-based integrated navigation and control for ROS enabled robots. Conference Proceedings - IEEE SOUTHEASTCON. https://doi.org/10.1109/SECON.2017.7925266

[8] Conner, D. C., Catherman, D., Enders, S., Gates, J., & Gut, J. (2018). Flexible Manipulation: Finite State Machine-based Collaborative Manipulation. Conference Proceedings - IEEE SOUTHEASTCON, 2018-April 1-8. https://doi.org/10.1109/SECON.2018.8478933

[9] Hou, J., Zhang, Y., & Rosendo, A. (n.d.). Mobile Manipulation Tutorial.

[10] http://wiki.ros.org/flexbe

[11] http://philserver.bplaced.net/fbe/

[12] Agis, R. A., Gottifredi, S., & García, A. J. (2020). An event-driven behavior trees extension to facilitate non-player multi-agent coordination in video games. Expert Systems with Applications, 155, 113457. https://doi.org/10.1016/j.eswa.2020.113457
[13] Marzinotto, A., Colledanchise, M., Smith, C., & Ogren, P. (2014). Towards a unified behavior Trees framework for robot control. Proceedings - IEEE International Conference on Robotics and Automation, 5420–5427. https://doi.org/10.1109/ICRA.2014.6907656
[14] Colledanchise, M., Almeida, D., & Ögren, P. (2019). Towards blended reactive planning and acting using behavior trees. Proceedings - IEEE International Conference on Robotics and Automation, 2019-May, 8839–8845. https://doi.org/10.1109/ICRA.2019.8794128
[15] Ding, H. Gao, R.X., Isaksson, AJ Landers, R.G., Parisini, T., Yuan, Y. State of Ai-Based Monitoring in Smart Manufacturing, and Introduction to Focused Section, IEE/ASME Transactions on Mecvhatronics, 25(5), 2020.
[16] Alexopoulos, K., Nikolakis, N., & Chryssolouris, G. (2020). Digital twin-driven supervised machine learning for the development of artificial intelligence applications in manufacturing. International Journal of Computer Integrated Manufacturing, 33(5), 429–439. https://doi.org/10.1080/0951192X.2020.1747642
[17] Malik, A. A., & Brem, A. (2021). Digital twins for collaborative robots: A case study in human-robot interaction. Robotics and Computer-Integrated Manufacturing, 68(September 2019), 102092. https://doi.org/10.1016/j.rcim.2020.102092
[18] Gkournelos, C., Kousi, N., Bavelos, A. C., Aivaliotis, S., Giannoulis, C., Michalos, G., & Makris, S. (2020). Model based reconfiguration of flexible production systems. Procedia CIRP, 86, 80–85. https://doi.org/10.1016/j.procir.2020.01.042
[19] Kousi, N., Gkournelos, C., Aivaliotis, S., Giannoulis, C., Michalos, G., & Makris, S. (2019). Digital twin for adaptation of robots' behavior in flexible robotic assembly lines. Procedia Manufacturing, 28, 121–126. https://doi.org/10.1016/J.PROMFG.2018.12.020
[20] Dröder, K., Bobka, P., Germann, T., Gabriel, F., & Dietrich, F. (2018). A machine learning-enhanced digital twin approach for human-robot-collaboration. Procedia CIRP, 76, 187–192. https://doi.org/10.1016/j.procir.2018.02.010
[21] Liu, J., Zhou, H., Liu, X., Tian, G., Wu, M., Cao, L., & Wang, W. (2019). Dynamic Evaluation Method of Machining Process Planning Based on Digital Twin. IEEE Access, 7, 19312–19323. https://doi.org/10.1109/ACCESS.2019.2893309
[22] Bilberg, A., & Malik, A. A. (2019). Digital twin driven human–robot collaborative assembly. CIRP Annals, 68(1), 499–502. https://doi.org/10.1016/j.cirp.2019.04.011
[23] Tsarouchi, P., Michalos, G., Makris, S., Athanasatos, T., Dimoulas, K., & Chryssolouris, G. (2017). On a human–robot workplace design and task allocation system. International Journal of Computer Integrated Manufacturing, 30(12), 1272–1279. https://doi.org/10.1080/0951192X.2017.1307524
[24] Andrianakos, G., Dimitropoulos, N., Michalos, G., & Makris, S. (2020). An approach for monitoring the execution of human based assembly operations using machine learning. Procedia CIRP, 86, 198–203. https://doi.org/10.1016/j.procir.2020.01.040
[25] Urgo, M., Tarabini, M., & Tolio, T. (2019). A human modelling and monitoring approach to support the execution of manufacturing operations. CIRP Annals, 68(1), 5–8. https://doi.org/10.1016/j.cirp.2019.04.052
[26] Maderna, R., Lanfredini, P., Zanchettin, A. M., & Rocco, P. (2019). Real-time monitoring of human task advancement. IEEE International Conference on Intelligent Robots and Systems, 433–440. https://doi.org/10.1109/IROS40897.2019.8967933

[27] Liu, H., & Wang, L. (2017). Human motion prediction for human-robot collaboration. Journal of Manufacturing Systems, 44, 287–294. https://doi.org/10.1016/j.jmsy.2017.04.009
[28] Cao, Z., Hidalgo, G., Simon, T., Wei, S. E., & Sheikh, Y. (2021). OpenPose: Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields. IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(1), 172–186. https://doi.org/10.1109/TPAMI.2019.2929257
[29] Wang, K., Lin, L., Jiang, C., Qian, C., & Wei, P. (2020). 3D Human Pose Machines with Self-Supervised Learning. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(5), 1069–1082. https://doi.org/10.1109/TPAMI.2019.2892452
[30] Martinez, J., Hossain, R., Romero, J., & Little, J. J. (2017). A Simple Yet Effective Baseline for 3d Human Pose Estimation. Proceedings of the IEEE International Conference on Computer Vision, 2017-Octob, 2659–2668. https://doi.org/10.1109/ICCV.2017.288
[31] Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., & Black, M. J. (2016). Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 9909 LNCS, 561–578. https://doi.org/10.1007/978-3-319-46454-1_34
[32] De Santis, A., Siciliano, B., De Luca, A., & Bicchi, A. (2008). An atlas of physical human-robot interaction. Mechanism and Machine Theory, 43(3), 253–270. https://doi.org/10.1016/j.mechmachtheory.2007.03.003
[33] Jung, Y., Kim, S., & Choi, B. (2016). Consumer valuation of the wearables: The case of smartwatches. Computers in Human Behavior, 63(2016), 899–905. https://doi.org/10.1016/j.chb.2016.06.040
[34] Villani, V., Sabattini, L., Battilani, N., & Fantuzzi, C. (2016). Smartwatch-Enhanced Interaction with an Advanced Troubleshooting System for Industrial Machines. IFAC-PapersOnLine, 49(19), 277–282. https://doi.org/10.1016/j.ifacol.2016.10.547
[35] Gkournelos, C., Karagiannis, P., Kousi, N., Michalos, G., Koukas, S., & Makris, S. (2018). Application of wearable devices for supporting operators in human-robot cooperative assembly tasks. Procedia CIRP, 76, 177–182. https://doi.org/10.1016/j.procir.2018.01.019
[36] Lotsaris, K., Gkournelos, C., Fousekis, N., Kousi, N., & Makris, S. (2020). AR based robot programming using teaching by demonstration techniques. Procedia CIRP, 97, 459–463. https://doi.org/10.1016/j.procir.2020.09.186
[37] Lotsaris, K., Fousekis, N., Koukas, S., Aivaliotis, S., Kousi, N., Michalos, G., & Makris, S. (2020). Augmented Reality (AR) based framework for supporting human workers in flexible manufacturing. Procedia CIRP, 96, 301–306. https://doi.org/10.1016/j.procir.2021.01.091
[38] http://gazebosim.org/
[39] https://moveit.ros.org/
[40] https://www.ros.org/

Appendix

Abbreviations

Table 8: Abbreviations

Abbreviation	Meaning
AI	Artificial Intelligence
AR	Augmented Reality
ASSISTANT	LeArning and robuSt decision SupporT systems for agile mANufacTuring environments
BOP	Bill Of Processes
BT	Behavior Tree
CMM	Coordinate Measuring Machine
CNC	Computer Numerical Control
DOA	Description Of Action
DT	Digital Twin
DTE	Digital Twin of Execution
DTW	Dynamic Time Wrapping
EU	European Union
FD	Fault Diagnosis and Fault Detection
FSM	Finite State Machine
HBD	Human Body Detection
HMM	Hidden Markov Model
HP	High Priority
HRC	Human Robot Collaboration
HRI	Human Robot Interaction
HTP	Human Task Prediction
IMU	Inertial Measurement Units
IO	Input/Output
KPIs	Key Performance Indicators
LP	Low Priority
LSTM	Long short-term memory
MPFDT	Manufacturing Process Failure Diagnosis Tool
NH	Nice to Have
pHRI	physical Human Robot Interaction
PLAT	PLC Log-Data Analysis Tool
PLC	Programmable logic controller
POV	Point Of View
RULP	Remaining Useful Life Prediction
SS	Show Stopper
WP	Work Package