Ref. Ares(2025)10774911 - 05/12/2025

AgiFlex

Tools, Requirements and KPIs

Henrik Saxén, Carl Haikarainen, Marius Thielen, Fabien Nkwitchoua, Maximilian Bernds, Thorsten Hauck, Lorenzo Vannini, Ismael Matino, Peter van den Broeke

5.12.2025

Project co-funded by the Horizon Europe programme

G.A. 101138813

Version	Date	Authors	Partners	Description
0.1	10.10.2025	H. Saxén, C. Haikarainen	ÅAU	First draft
0.2	30.10.2025	Marius Thielen	DH	First draft chapter 3
0.3	31.10.2025	F. Nkwitchoua	BFI	First draft chapter 2
0.4	20.11.2025	P. van den Broeke	TSNT	First draft chapter 5
0.5	26.11.2025	I. Matino, L. Vannini	SSSA	First draft chapter 4
1.0	5.12.2025	H. Saxén, C. Haikarainen, M. Thielen, F. Nkwitchoua, M. Bernds, T. Hauck, L. Vannini, I. Matino, P. van den Broeke	ÅAU, DH, BFI, SSSA, TSNT	Submitted version

Table of content

1 Introduction

This document summarizes the outcomes of Tasks 1.1 and 1.2 in the AgiFlex project, i.e. "Current tools and future requirements on different time horizons" and "Specification of plant performance KPIs and relevant data sources".

The document first presents some background on multi-agent systems and an overview of available tools and frameworks for implementing them, as well as motivations for initial choices regarding system frameworks in the project in Section 2. Section 3 discusses demands and possibilities of AgiFlex from the perspective of plant management in steel industry transitions towards decarbonised steel production. Section 4 complements the previously published Data Management Plan (D7.3) in defining required data sources for AgiFlex implementation. The document finishes with definitions and explanations of key performance indicators for measuring system performance in the project in Section 5 and some concluding paragraphs in Section 6.

2 Current available tools

This section presents a selection of tools currently available for implementing multi-agent systems and provides some basic concepts for the implementation of the two frameworks for the use cases of BFI and TSNT.

2.1 Background on Multi-Agent Systems

A **Multi-Agent System (MAS)** is composed of several autonomous and interactive entities, known as *agents*, that operate within a shared environment to achieve individual or collective objectives. Each agent possesses the ability to **perceive**, **reason**, **communicate**, and **act** based on its internal goals and the state of its surroundings. MAS architectures have demonstrated significant effectiveness in managing **distributed, dynamic, and complex problems**, particularly in cases where **centralised control** is either infeasible or suboptimal (Wooldridge, 2009).

In recent years, MAS has been increasingly combined with **Digital Twin (DT)** technologies to form intelligent and interconnected industrial ecosystems (Ricci et al., 2022; van Dyck et al., 2023). In such hybrid frameworks, each agent can embody the digital representation of a physical asset (e.g., a furnace, rolling mill, or energy storage system), enabling **real-time synchronisation**, **scenario simulation**, **predictive analytics**, and **adaptive decision-making** across the system.

Figure 2.1 shows a general overview of MAS integrating digital twins of industrial units (e.g. furnace, rolling mill, steel work). The Infrastructure/Communication Layer enables data or information exchange via communication channels such as OPC UA, MQTT, TCP/IP, and REST API for interaction between hardware, software, agents and digital twins.

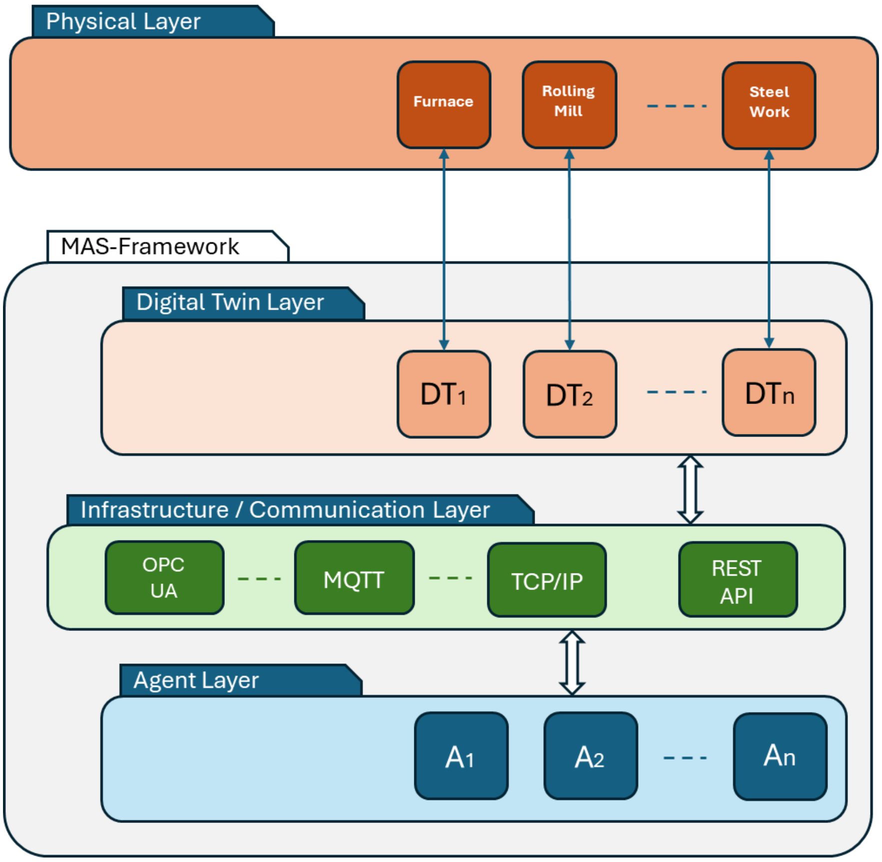

Figure 2.1. General overview of the MAS Framework with Digital Twins

2.1.1 Fundamental Properties of Multi-Agent Systems

Multi-agent systems are characterised by several fundamental properties that enable robust, scalable, and intelligent coordination:

- **Autonomy:** Agents operate independently, making decisions without external control, based on their internal state and objectives.
- **Social Ability:** Agents communicate through predefined protocols, engaging in negotiation, cooperation, or competition to achieve local or global goals.
- **Reactivity:** Agents can detect environmental changes and respond in real-time, ensuring system adaptability to disturbances or uncertainties.
- **Proactiveness:** Agents exhibit goal-oriented behaviour, taking initiative to achieve objectives rather than reacting solely to events.
- **Adaptability and Learning:** Through reinforcement, feedback, or machine learning mechanisms, agents adjust their strategies to improve future performance.
- **Scalability and Robustness:** MAS architectures allow seamless addition or removal of agents without compromising global system stability, making them suitable for large-scale optimisation problems.

2.1.2 Coordination and Optimisation under Constraints

When applied to optimisation problems, MAS often operate under **physical**, **operational**, or **resource constraints**, which necessitate efficient **coordination** and **distributed optimisation** strategies. Several paradigms are used to achieve this:

- **Constraint Satisfaction and Propagation:** Agents iteratively exchange feasible domains and propagate local constraints to ensure the global consistency of solutions.
- **Distributed Optimisation and Consensus Mechanisms:** Techniques such as the Alternating Direction Method of Multipliers (ADMM) enable agents to converge towards an optimal global solution while preserving autonomy.
- **Game-Theoretic Coordination:** Agents act as rational players that seek equilibrium solutions where individual objectives and shared constraints are simultaneously satisfied.
- **Market-Based Mechanisms:** Agents negotiate and trade limited resources (e.g., energy, material flow) within virtual markets to reach an efficient equilibrium.
- **Hierarchical and Hybrid Control:** A supervisory layer ensures global objective alignment and constraint satisfaction, while local agents optimise their subsystems independently.

2.1.3 MAS for Constrained Industrial Optimisation

In industrial environments, MAS integrated with Digital Twins are particularly suited to **real-time or offline constrained optimisation**. Each agent represents a physical process or subsystem that operates within local boundaries (e.g., pressure, temperature, throughput) while contributing to a **global optimisation objective**. Examples include:

- **Energy and material flow balancing** between interconnected production units.
- **Scheduling and resource allocation** under capacity and safety limits.
- **Multi-objective optimisation**, balancing energy efficiency, cost, and production quality.
- **Dynamic constraint management**, where agents adapt continuously to maintain feasible and optimal operating conditions.

By leveraging decentralised intelligence, communication, and coordination, MAS-based architectures enable **collective optimisation** in complex industrial systems. This approach enhances system flexibility, resilience, and global performance while maintaining compliance with operational constraints in real time.

2.2 Overview of Available MAS Frameworks

Several MAS frameworks and platforms have been developed to facilitate the design, simulation, and deployment of agent-based applications. Table 2.1 summarizes key characteristics of widely used MAS tools.

Table 2.1. Overview of common MAS frameworks

Framework	Language	Description	Typical Applications
JADE (Java Agent DEvelopment Framework) (Bellifemine, 2007).	Java	FIPA-compliant platform for distributed communication and agent lifecycle management.	Telecommunications, distributed AI.
Jason (Bordini, R. H., Hübner, J. F., & Wooldridge, M. 2007).	Java (AgentSpeak)	Belief–Desire–Intention (BDI) architecture; suitable for reasoning/cognitive agents.	Reasoning agents, AI research.
MESA (https://mesa.readthedocs.io/latest/) (Masad, 2015).	Python	Lightweight ABM framework with simple web visualization and strong data-science integration.	Digital Twins, social systems, industrial prototyping.
SPADE (https://spadeagents.eu/)	Python	Asynchronous messaging (XMPP) for distributed real-time MAS applications.	IoT, smart grids, robotics.
GAMA Platform	Java	Graphical environment for spatial/geographical agent-based modeling.	Urban, ecological, environmental simulations.
Repast	Java / C++ / .NET	High-performance toolkit for large-scale, parallel ABM.	Computational science, social modeling.
NetLogo	Proprietary	Educational and accessible environment for small-to-medium ABMs.	Education, teaching, conceptual modeling.

AgentPy	Python	Open-source library for the development and analysis of ABMs; integrates model design, simulation, experiments, and data analysis in Jupyter environments.	Research prototypes, interactive modeling in Jupyter.
FLAME	C / C++ (HPC)	Generic agent-based modeling platform with XML model specification; targets large-scale/HPC execution.	Biology, economics, large-scale simulations on clusters.
FLAME GPU	CUDA C++ (Python bindings)	GPU-accelerated ABM for very large agent counts and real-time/near-real-time execution.	Crowds, transportation, biology, large-scale complex systems.
Agents.jl	Julia	High-performance ABM library for Julia with composable, type-stable models and interactive tooling.	Research, fast prototyping with performance emphasis.
mango	Python	Modular agent simulation framework focusing on message-based agents and scalable distributed simulation.	Research frameworks emphasizing modular and maintainable MAS.

2.3 Developing a Custom MAS Framework from Zero

While existing tools significantly reduce development time, they can impose limitations that restrict adaptability, especially in **industrial environments** where **Digital Twins**, **real-time data**, and **custom protocols** are involved.

To achieve full control, some researchers and developers prefer to **design MAS frameworks from scratch**, following modular and service-oriented principles.

2.3.1 Flexibility

Developing a MAS from zero enables the definition of:

- Custom **communication protocols** (e.g., MQTT, TCP/IP, REST, ZeroMQ).
- Tailored **decision architectures** (reactive, cognitive, or hybrid).
- Direct integration with existing **Digital Twin** infrastructures or SCADA systems. This approach allows the MAS to align perfectly with specific industrial requirements and data pipelines.

2.3.2 Modularity

A custom MAS can be designed as a **microservice architecture**, where each agent, Digital Twin, and supervisory unit is a separate, independent module. This modularity facilitates:

- Easier maintenance and debugging.
- Independent deployment and updates.
- Collaboration among multiple development teams.

2.3.3 Scalability

From an architectural standpoint, scalability can be achieved by deploying agents across distributed computing environments or cloud services. This allows:

- Dynamic addition/removal of agents at runtime.
- Horizontal scaling using containerization technologies (e.g., Docker, Kubernetes).
- Integration with edge devices for real-time control.

2.3.4 Integration with Digital Twins

In industrial contexts, the most promising configuration is a **MAS–Digital Twin hybrid system**, where:

- Each agent either represents or interacts with a Digital Twin.
- Real-time synchronization ensures **data consistency** between the physical and virtual layers.
- The MAS supervises and optimizes operations (e.g., energy distribution, predictive maintenance, or quality control).

Figure 2.2 illustrates an example of a modular Multi-Agent System (MAS) architecture integrating Digital Twins. The MAS-Environment is composed of several modules, including the User Interface (UI-Module), Supervision Module for communication management, Data Module for data storage, MAS-Module for coordination and interaction, Digital Twin (DTs-Module) for process modelling, and Optimization Module (Opt-Module) for adaptive decision-making and performance improvement.

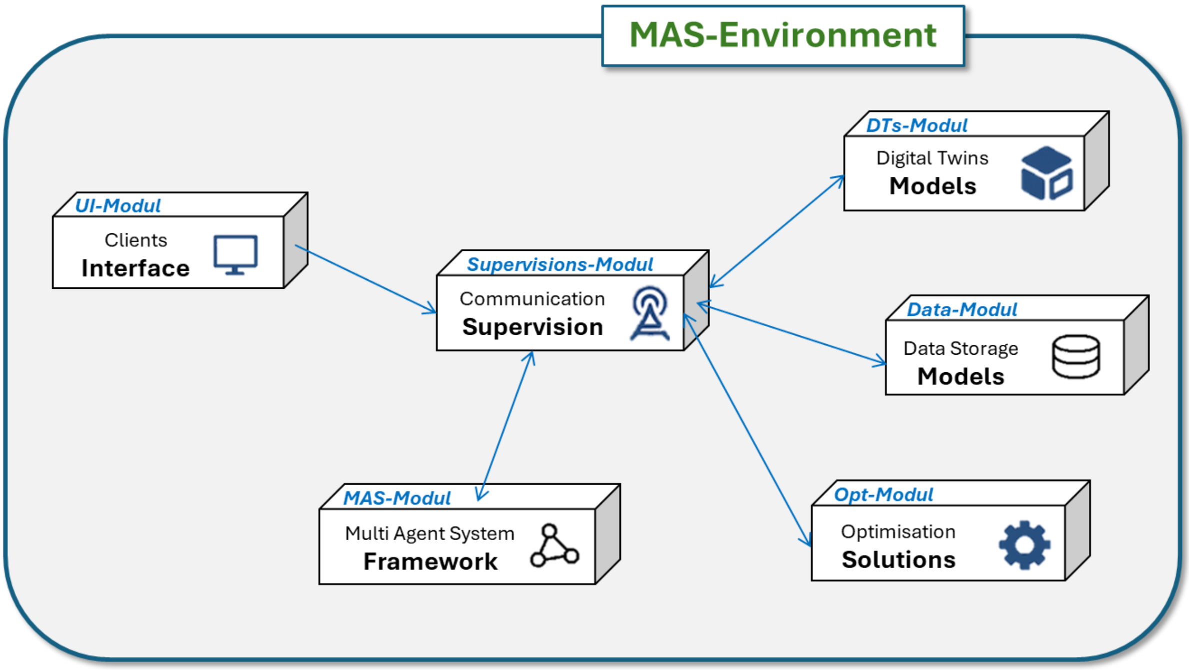

Figure 2.2. Example architecture of a modular MAS integrating Digital Twins

2.4 Comparative Evaluation

Table 2.2 provides a comparative overview between **existing MAS frameworks** and **custom-built MAS architectures**.

Table 2.2: Comparative evaluation of MAS Framework implementation approach

Approach	Advantages	Limitations
Existing MAS Frameworks (JADE, MESA, SPADE, etc.)	- Ready-to-use infrastructure - Simplified communication setup - Active user communities	- Limited flexibility for industrial integration - Fixed architectures - Language/platform dependencies - Possible licence fees
Custom MAS Framework (from zero)	- Maximum flexibility and modularity - Seamless integration with Digital Twins and industrial protocols - Scalable for offline or real-time control	- Higher initial development effort - Requires strong software engineering discipline

2.5 MAS Language Selection

Independent of the MAS implementation approach, the first step consists of selecting a suitable programming language according to the defined objectives and the available software engineering capabilities.

Table 2.3 presents an example of answers to questions guiding the choice of the most appropriate language for implementing a specific use case with a customised MAS framework at BFI.

Table 2.3: Criteria and questions guiding the selection of a suitable MAS programming language.

Section	Question	Options
Project Context	What is the main goal of the multi-agent platform?	☐ Optimisation of complex calculations. ☐ Collaboration of multiple agents worldwide. ☐ Interactive results analysis. ☐ Other (specify):
Project Context	What types of models will the platform handle?	☐ Surrogate models. ☐ Regression models. ☐ Deep neural networks. ☐ Gaussian processes. ☐ Other (specify):
Project Context	What is the estimated complexity of the data to be processed?	☐ Small datasets (less than 10,000 points). ☐ Medium datasets (10,000 to 1 million points). ☐ Large datasets (over 1 million points).
Project Context	What is the frequency of communication between agents worldw	☐ Frequent (every second). ☐ Moderate (every minute). ☐ Rare (hourly or daily).
Performance and Flexibility Criteria	What is the priority level for computational performance?	☐ Very high (computations must be extremely fast). ☐ Medium (computations should be optimised but not critical). ☐ Low (priority is flexibility, not speed).
Performance and Flexibility Criteria	What is the priority level for flexibility in adding or modifying mod	☐ Very high (developers must easily modify and test models). ☐ Medium (modifications are occasional). ☐ Low (models are rarely modified).
Performance and Flexibility Criteria	Should agents be capable of learning (machine learning)?	☐ Yes. ☐ No. ☐ Not yet defined.
Technical Environment and Constrain	What programming languages are currently mastered by the tean	☐ Java. ☐ Python. ☐ C++. ☐ Julia. ☐ Other (specify):
Technical Environment and Constrain	Which platforms should be supported?	☐ Windows. ☐ Linux. ☐ macOS. ☐ Cloud (AWS, Azure, GCP). ☐ Other (specify):
Technical Environment and Constrain	What is the tolerance for communication failures or interruptions	☐ Very low (platform must be highly resilient). ☐ Medium (interruptions are tolerable if agents resume tasks). ☐ Low (system can restart without major issues).
Data Management and Analysis	How do you want to analyse the results?	☐ Through a graphical interface (interactive dashboards). ☐ Through automatic system-generated reports. ☐ Other (specify):
Data Management and Analysis	What volume of output data will the agents handle?	☐ Low (simple textual data). ☐ Medium (structured data such as tables or JSON). ☐ High (large data such as images or heavy files).
Collaboration and Communication	Do teams work in different time zones globally?	☐ Yes. ☐ No.
Collaboration and Communication	Which collaboration tools are already being used?	☐ Git/GitHub/GitLab. ☐ CI/CD tools. ☐ Project management tools (Trello, Jira). ☐ Other (specify):
Final Selection Criteria	What are the most important criteria for choosing the language?	☐ Execution speed. ☐ Ease of integration into distributed systems. ☐ Prototyping and experimentation flexibility. ☐ Community support and documentation. ☐ Compatibility with existing infrastructures. ☐ Other (specify):

2.6 Practical Examples of MAS Implementation

This section gives some basic examples of the implementation of a MAS using an existing or a custom-built framework.

2.6.1 Industrial Energy Optimization

Description: Agents represent Digital Twins of industrial units such as blast furnaces, rolling mills, and reheating furnaces (https://mesa.readthedocs.io/latest/examples.html)

Implementation Details:

Aspect	**Details**
Platform	Custom Python MAS using Flask (web interface), MQTT (message exchange), and MESA (agent logic).
Agent Types	EnergyAgent (monitors consumption), ProcessAgent (represents unit twin), SupervisorAgent (optimizes energy).
Communication	MQTT broker (e.g., Mosquitto) handles asynchronous message passing. JSON encodes variables like power, CO_2_emission, status.
Data Flow	1. ProcessAgent publishes consumption. 2. SupervisorAgent compares against budget. 3. Flask dashboard visualizes results.
Integration	Possible link to OPC-UA or Modbus for live industrial data.
Outcome	Real-time optimization and decentralized decision-making.

2.6.2 Smart Grid Management

Description: Agents coordinate distributed energy resources and consumers for efficient grid balancing.

Aspect	**Details**
Platform	SPADE or custom asynchronous MAS using asyncio and aiohttp.
Agent Types	ProducerAgent (renewables), ConsumerAgent (loads), GridAgent (market coordinator).
Negotiation	Contract Net Protocol or auction-based for demand-response.

Communication	XMPP or MQTT with JSON or FIPA-ACL message format.
Integration	Renewable or SCADA data via OPC-UA.
Outcome	Distributed negotiation and adaptive load balancing.

2.6.3 Predictive Maintenance Systems

Description: Agents detect anomalies, update digital twins, and trigger maintenance actions automatically.

Aspect	**Details**
Platform	Custom MAS integrated with sensor-based Digital Twins via OPC-UA.
Agent Types	SensorAgent (collects data), ModelAgent (analyzes data), MaintenanceAgent (schedules actions).
Architecture	OPC-UA client connects to PLCs. ModelAgent compares live vs. predicted data using ML/GEKKO models.
Data Flow	1. SensorAgent transmits data. 2. ModelAgent computes health index. 3. MaintenanceAgent triggers alert.
Integration	Database (SQLite/PostgreSQL) and visualization via Flask or Grafana.
Outcome	Predictive, not reactive, maintenance reduces downtime.

2.6.4 Logistics and Supply Chain Coordination

Description: Agents represent suppliers, transporters, and warehouses, coordinating to minimize costs and delays.

Aspect	**Details**
Platform	JADE or MESA-based Python MAS.
Agent Types	SupplierAgent (stock provider), TransportAgent (delivery), WarehouseAgent (inventory), CoordinatorAgent (optimizer).
Algorithm	Dijkstra for route planning, GEKKO/Pyomo for supply optimization.
Communication	REST API or MQTT; JSON defines order, route, and ETA messages.

Data Flow	1. WarehouseAgent triggers order. 2. SupplierAgent confirms. 3. TransportAgent executes delivery.
Outcome	Real-time coordination and cost reduction.

2.7 Common Technological Stack Across All Examples

Different parts of MAS implementation utilise different existing tools, libraries and frameworks. The following are some of the technologies used across all examples:

Layer	Tools / Libraries
Communication	MQTT, SPADE, Flask, REST API
Modeling	MESA, Pyomo, GEKKO, scikit-learn
Data	SQLite, PostgreSQL, OPC-UA
Visualization	UI (Java, C++, etc.), Flask Dashboard, Chart.js, Grafana
Deployment	Docker, Kubernetes, ...

2.8 Summary and Discussion

The selection of an MAS framework depends on the **application domain**, **desired flexibility**, and **integration requirements**.

For educational and research purposes, tools such as **MESA**, **NetLogo**, or **GAMA** provide an accessible entry point.

For distributed and industrial-scale systems, **JADE**, **SPADE**, or **Repast** offer robust communication infrastructures.

However, when **maximum flexibility, modularity, and scalability** are required-especially for integrating **Digital Twins**, **real-time data**, and **industrial protocols-**developing a **custom MAS framework** is the optimal solution.

This approach enables:

- Fine-grained control of communication and decision-making layers,
- Seamless integration with physical and virtual environments, and
- Long-term adaptability to evolving industrial processes.

3 Requirements for plant management

The steel industry is undergoing a profound transformation driven by the urgent need to decarbonize production processes. This transition introduces unprecedented complexity into plant management, as conventional systems are gradually replaced or operated in parallel with emerging low-carbon technologies. In this context, traditional planning tools—often based on static flowsheets and deterministic assumptions—are no longer sufficient to support decision-making under volatile market and energy conditions.

We identify the following key challenges in modern plant management:

1. **Complex Production Chains**: Integrated steelworks increasingly operate hybrid configurations combining legacy units (e.g., BF-BOF) with new technologies (e.g., DR-EAF). Managing material and energy flows across these mixed chains requires dynamic, flexible planning tools. Even during production planning, we must carefully evaluate the costs associated with choosing one route over another. This, combined with customer constraints, makes it a highly complex planning challenge.

2. **Uncertainty and Volatility**: Fluctuating energy prices, variable availability of (renewable) electricity, and evolving carbon pricing schemes introduce significant uncertainty. Static models cannot adequately capture these dynamics, leading to suboptimal or risky investment and operational decisions.

3. **Scenario Planning and Investment Strategy**: Long-term strategic decisions—such as when and how to invest in new technologies or when to plan downtimes —must be made under uncertain future conditions. Existing tools lack the capability to evaluate multiple scenarios with probabilistic inputs and structural flexibility.

4. **Demand-Side Response (DSR)**: As steel plants become major consumers of electricity, their ability to adapt to grid conditions becomes critical. Current systems, however, do not support flexible operation strategies that align production with energy market signals.

5. **Process Integration and Emissions Management**: The transition to the EAF-route affects internal gas and energy flows, requiring re-optimization of process integration to minimize emissions. This includes the valorisation of by-product gases and the integration of CCS/CCU technologies.

6. **Multi-Level Planning Requirements**: Plant management spans strategic (years), tactical (months), and operational (days to hours) horizons. Each level demands tailored models and optimization strategies, which are rarely integrated in existing solutions.

To address these challenges, the AgiFlex project introduces a novel agent-based optimization framework that combines digital twins of process units with distributed decision-making algorithms. This system enables:

- Holistic optimization of plant configurations and operations.

- Scenario-based planning under uncertainty.
- Real-time adaptation to market and energy fluctuations.
- Integration of environmental and economic objectives.

By leveraging multi-agent systems and scalable digital models, AgiFlex provides a robust and flexible tool for managing the complexity of steel production during the green transition.

As the effectiveness of AgiFlex's optimization algorithms depends on the accuracy and completeness of input data, collection of available data was started in two existing steel plants, Dillinger and Tata Steel Netherlands, to feed the digital twins and agents. This data must, however, be collected under brownfield conditions, which means that data may be stored in isolated databases with limited interoperability, accessible only via manual exports or non-standard APIs and/or inconsistent in granularity, frequency, and timestamping in many different systems. Data quality must also be addressed at this point. In brownfield environments, there are often sensor drift and calibration issues leading to inaccurate readings, as well as missing or corrupted data, especially during maintenance, downtime or start-up. In contrast to that, AgiFlex's operational planning modules require high-quality data to support dynamic decision-making. To mitigate these problems, good data availability must be secured, for example by storing the data in a well-accessible database, coupled with robust data preprocessing techniques to secure a high enough data quality for the AgiFlex-system.

Having addressed the plant data, markets must also be taken into account. In the current market we see semi-regular fluctuations that must be considered within AgiFlex. Here, an accurate market prediction will enable a great potential in reducing energy cost, making this essential for achieving good results.

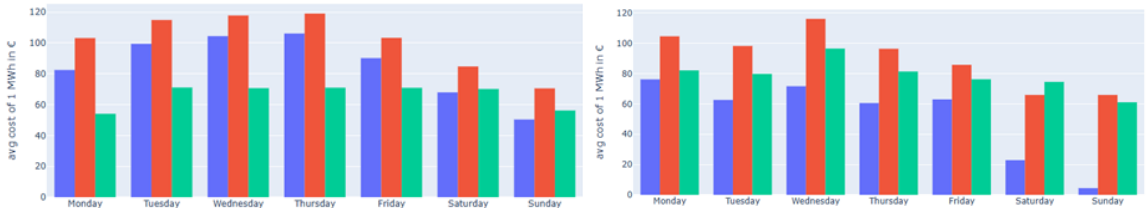

Figure 3.1: Average costs of one MWh of electricity at the German-Luxembourg market during early shift (blue), late shift(red) and night shift (green) per weekday in 2024. Left: winter (January - April & October - December), right: summer (May -September).

As seen in Figure 3.1, in winter electricity is more expensive making it a potential time for maintenance to reduce costs. Further, in the summer during the early shift the electricity is on average cheapest, while it during the late shift is more expensive. Adapting production to this rhythm will greatly increase the potential of AgiFlex, so a good market prediction will be needed.

Lastly, the way results are presented to the users must be carefully considered. As different users have different requirements and goals, the results should be made available in different forms. While a planner may be interested in scenario simulation, an energy manager may be more interested in the operational real-time application, preferring a visual representation of the results. Thus, different user groups can be considered.

Strategic planners and plant managers engage with AgiFlex at a high level of abstraction. Through scenario modeling, AgiFlex allows these users to evaluate the economic and environmental implications of various decarbonization pathways and downtime strategies. The system’s ability to simulate future plant states under uncertain boundary conditions—such as energy prices or carbon penalties—provides a robust foundation for strategic planning and risk assessment.

Energy managers form another critical group of users. Their role is to oversee the procurement, distribution, and optimization of energy carriers, including electricity, natural gas, hydrogen, and internally generated process gases. AgiFlex supports these tasks by offering tactical planning modules that align energy supply with production demand, while also enabling scenario-based analysis of energy dispatch strategies. This functionality is particularly valuable in the context of fluctuating energy markets, where demand-side response and flexible load management can yield significant cost and sustainability benefits.

The AgiFlex framework can potentially also be further developed towards more short-term management, to be used by production engineers who are responsible for the day-to-day operation of key process units such as blast furnaces, electric arc furnaces, and direct reduction reactors. These engineers could benefit from real-time planning capabilities, which allow them to simulate and evaluate alternative production sequences, optimize batch scheduling, and respond dynamically to changes in energy availability or raw material quality. Digital twins could provide a transparent and data-driven representation of each unit’s behaviour, enabling engineers to make informed decisions that enhance process efficiency and reduce emissions. However, the focus of AgiFlex is on the strategic and tactical planning levels, while more short-term management can be seen as a potential future feature to develop.

Finally, IT and automation specialists play a pivotal role in the deployment and maintenance of AgiFlex. They are responsible for integrating the system with existing MES, ERP, and control platforms, ensuring secure data exchange and system stability. AgiFlex’s modular architecture and use of standardized protocols such as OPC UA facilitate this integration, while its protected-mode operation ensures that optimization activities do not interfere with plant control systems during validation phases.

Collectively, these user groups form a multi-disciplinary ecosystem that leverages AgiFlex to improve operational performance, support strategic transformation, and enhance the resilience of steel production under the pressures of decarbonization and energy transition. By aligning the system’s capabilities with the specific needs and responsibilities of each user, AgiFlex not only empowers individuals in their roles but also contributes to the broader organizational goal of sustainable and flexible steelmaking.

To be able to secure good access to AgiFlex for all users, it will have to be made available on high-performance workstations, which will be either remotely accessible or able to communicate with the users' devices in order to secure the usage of AgiFlex.

The different user groups will also be the experts that are needed to validate results, so the AgiFlex-system must be reliable and easy to use for all those users. The real-world system will be compared with results from AgiFlex in an intense exchange with these experts, collecting important feedback to reconsider results and reoptimize the AgiFlex tools.

In conclusion, AgiFlex must be very flexible in data operation and usage as well as in providing its results, whilst integrating the users into the development of AgiFlex, incorporating their needs and thus increasing their acceptance of the new tool.

4 Data sources and harmonization procedure

Project implementation requires data from the demonstrator plants. Indeed, data are fundamental for the AgiFlex project, both for model development and for foreseen investigations.

As already explained in D7.3 "Data Management Plan" (DMP) the expected data to be exploited in the project are numerous and the sources for their collection and use are different.

Focusing on a more practical point of view and expanding on what was already explained in the DMP, data can be clustered in the following categories: **operational data, market or outside plant boundaries data, plant specifications, roadmap information, and data from plant foresight analysis. Each of these data comes from different sources**.

Below is a comprehensive list describing each data category and providing an overview of related sources:

- **Operational data:** Operational data refer to generally dynamic data coming from process operations and management and are the information mainly used in AgiFlex for model development and usage. For any process, it is essential to know what data are available, how many features have to be included and how complex the model should be. Knowledge of inputs and outputs is also fundamental. Demonstrator plants are the main sources of these data. More specifically, they come from distributed control systems (DCS), programmable logic controllers (PLC), field instrumentation and sensors, and data acquisition systems (SCADA). Further sources are also the manufacturing execution systems for data more related to the production traceability, raw materials consumptions, yields, by-products and wastes. Other operational data are available also from discontinuous and laboratory analyses (e.g. for physical and chemical tests).
- **Market or outside plant boundaries data:** Data about the market or outside plant boundaries are collected to understand the external environment in which a system or service operates. They refer to dynamic data of raw material and energy prices and markets, steel prices and demands but can also include macroeconomics data, international data and competitor information. The aim is to capture information that is not generated internally, but comes from the market, customers, competitors, or the physical infrastructure outside the organization, which constitute the sources of these data.
- **Plant specification:** A plant specification refers to a detailed description of main features of a plant, operational unit, system, or infrastructure both current and future. These are static data and define the design, size, capacity, included equipment and auxiliaries, processes, operation conditions, specifications and constraints,

performance and requirements that the plant must meet. A plant specification does more than describe the physical assets and operational constraints; it also provides reference points for performance. This means it defines target values, ideal operating conditions, or reference parameters that the plant should achieve. It is remarkably important to also know input and output nature. Sources of these kinds of data are mainly the demonstrator plants or plant/technology providers but also service infrastructure managers.

- **Simulated data:** Simulated data refer to data generated synthetically through models or computational simulations rather than collected from real world data measurements. They come from internal models of industrial and research partners involved in AgiFlex.
- **Roadmap information:** Roadmap information aims at describing the future developments and planned evolution of the plant, of its configuration and of the related scenarios simulated with developed models. It is used as a blueprint to specify all the info required to consider the transition towards C-lean processes. Main sources of these data are both internal facilities plans as well as European commission and sector platforms documents.
- **Data from plant foresight analysis:** These data and information come from internal analyses done by steelworks companies (i.e. demonstrator plants) about possible developments at a plant. They are complementary to the other available data and must be considered for appropriate design of scenarios and optimizations in the project.

As evident, the different data types, sources and collection methods make related management and usage complex. Moreover, it is necessary to have a common general framework to harmonize model development and use, and for optimization, computations and KPI calculations. Therefore, it is considered fundamental to share a common modus operandi and a common data structure framework to increase clarity, to enable data sharing with project partners, and to speed up model development and management.

Below, the identified solutions are described to achieve maximum compliance and ensure the project runs as smoothly as possible. These procedures can be considered complementary to DMP.

Each process should have a specific and homogeneous data table. The table should have 7 columns and 6 rows. Each row represents a data category, and each column represents a data specification.

The 6 data categories are the ones previously listed and described.

The 7 columns representing data specifications are: **Aim, Minimal Data, Preferred Data, Origin, Format, UoM, Frequency.**

Data specifications ensure that everyone modelling the process and using the models agrees on data characteristics, eliminating ambiguity.

Each row, or equivalently, each data category, will serve as a specification and description.

It is vital for modelling data to have a well-structured data table that meticulously defines attributes like Unit of Measure (UoM), Frequency, and Origin of Data and so on because these definitions directly determine the mathematical validity and predictive power of any quantitative model.

Below is a comprehensive list describing each data specification:

- **The Aim:** Identifies the precise purpose of the dataset. This avoids collecting irrelevant information and ensures that the data aligns with the analytical or operational objectives. When the aim is explicit, data engineers and downstream models do not rely on assumptions.
- **Minimal Data:** Defines the set of attributes without which the dataset loses meaning. This protects the pipeline from incomplete records and supports robust validation. Whenever a minimal attribute is missing, performance degradation can be diagnosed immediately. Clearly, Data security and Data privacy need to be considered. For the sake of clarity, and to give an example for operational data or for plant specifications, minimal data refer to data that are surely available at each steelworks (e.g. raw materials feed, steel production, energy consumption of plant areas, temperatures trend, off-gas production, unit capacity, critical constraints, emissions constraints).
- **Preferred Data:** Specifies additional variables that enhance accuracy, generalization, or interpretability. Although not strictly required, these variables strengthen statistical models by providing richer context. In many applied systems the difference between acceptable and excellent performance lies precisely in the inclusion of preferred variables. Clearly, Data security and Data privacy need to be considered. Again, for the sake of clarity, and as an example for operational data or for plant specifications, preferred data refer to data that are available only in some steelworks (e.g. detailed off-gas composition, specific energy consumption of each unit, by-products production).
- **Origin documents:** The source of each field. This is fundamental for traceability and reproducibility. When the provenance of a variable is known, its reliability, uncertainty, and potential biases can be assessed. This prevents silent data drift caused by unnoticed source changes.
- **Format:** Establishes how the data is encoded. Consistent formats eliminate ambiguity and prevent errors during parsing, transformation, and modelling.
- **Unit of Measure** is indispensable for meaningful numerical interpretation. A model trained on meters cannot be expected to interpret centimetres or feet without explicit conversion. Misaligned units create scaling discrepancies that severely affect model quality.
- **Frequency** determines the expected/actual temporal resolution of the data. Time dependent models rely on this assumption. A mismatch between expected and actual frequency introduces irregular gaps that compromise model stability.

When all these elements are formally defined, the data becomes predictable and internally coherent. This prevents ambiguity and enhances robustness.

By combining data specification and data category it is possible to obtain a fully comprehensive and coherent data format. In more practical terms, whenever a process needs to be modelled, a 5 x 7 table is defined and shared among partners. To clarify the process an example regarding EAF is included in the project shared folder.

5 Key performance indicators

Key performance indicators (KPIs) defined in the project are used to measure system performance in terms of, e.g., ecological and economic impacts, and also function as objectives for the agent systems.
To identify suitable KPIs, the most common steelwork decarbonization pathway is considered: future operation of many steelworks, such as those at the Tata Steel site in IJmuiden (Netherlands) or at Dillinger in Dillingen (Germany), will pursue a replacement of initially one of the blast furnaces with a gas-based direct reduction process and an electric arc furnace.

The main objective is to realize a major decrease in the overall CO_2 emission of the site (e.g. in the range of 4 to 5 Mton per annum for the Tata case). The time frame for establishing the new configuration by combining a blast furnace and a basic oxygen furnace, the BF - BOF route, with a direct reduction process and an electric arc furnace, the DRP - EAF route, is targeted to be from 2030 onward at Tata, a bit sooner at Dillinger.

Definition of KPIs

In terms of decarbonization, several parameters, KPIs, have been identified to be optimized for the production of steel products per year while at the same time achieving the threshold reduction in emissions (e.g. at least 4 Mton CO_2 per year for the Tata case). Additional aspects will include an increase in the use of scrap and maximizing the use of renewable feedstock. An evaluation will be made of the optimum combination for the use of renewable electricity and the use of hydrogen, methane and biochar, as reducing agents and as energy sources for generating mainly heat.

As an example, Figure 5.1 gives an overview of the expected future configuration at Tata Steel in IJmuiden. The current steel production at Tata Steel is based on the use of two blast furnaces, and the main step for the next decade, the years 2026 till 2035, is to have a new configuration, combining a BF - BOF route and a DRP - EAF route. To compare the performance of different steel plant configurations and operational schemes, the following KPIs will be considered in the project:

KPI	Additional description
Overall CO_2 reduction	For instance, a determined annual target can be part of a transition plan
Production rate	For instance, maintaining the production rate of steel after a transition can be highly relevant
DRI production rate	DRI production rates can reflect the state of a transition

Scrap use	Increased use of recycled steel scrap can reduce the need for virgin resources and energy
Contribution margin	The economic performance of a steel plant is a central optimisation parameter
Requirement of raw materials and energy sources	Material and energy streams to a steel plant are a tangible measure of resource use
Circularity	Steel plant internal use of material and energy streams reflects plant efficiency
Use of renewables	Use of renewables can be increased both in terms of material and energy sources

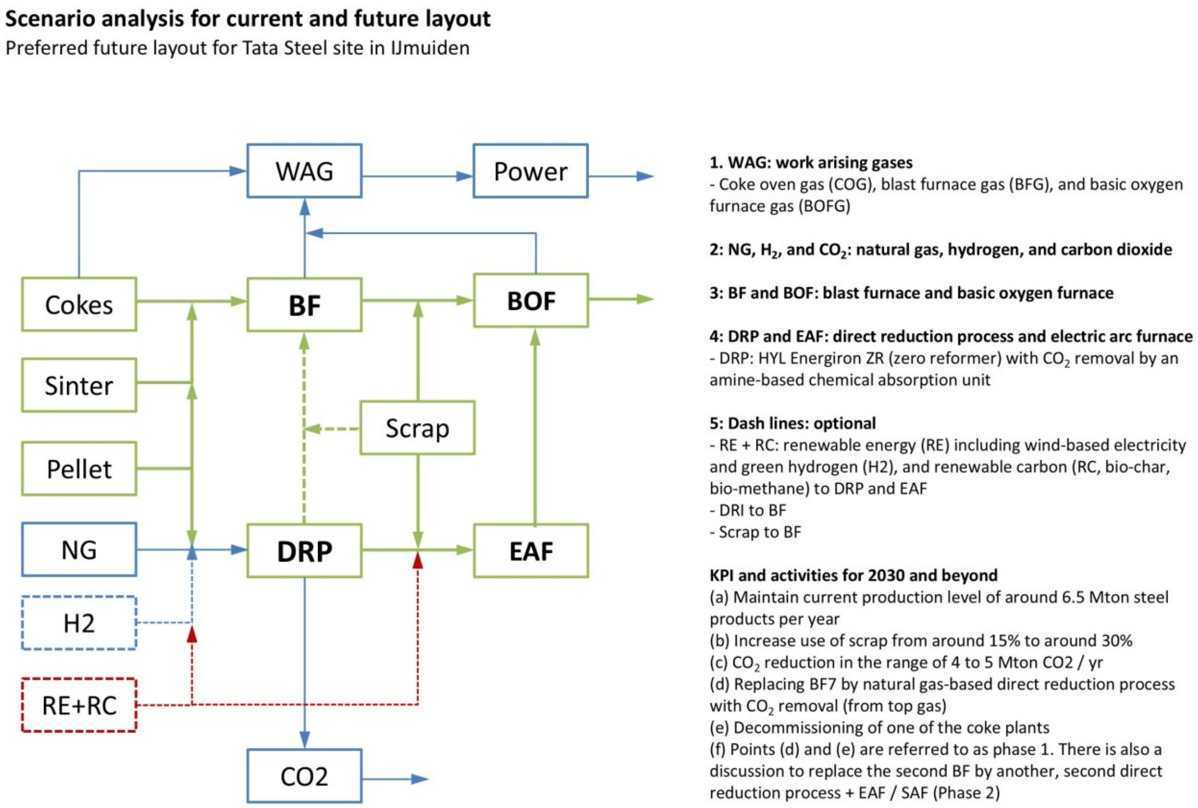

Figure 5.1. Simplified overview of the anticipated future configuration of the integrated steelworks at Tata Steel in IJmuiden (Netherlands).

Evaluation of site-specific KPIs for transition pathways

Various aspects of evaluating KPIs are site-specific, and also time-dependent as technologies and market conditions continuously evolve. Within the context of the AgiFlex project an overarching aspect is to minimize the carbon footprint of the products produced in the individual four main units, the BF, BOF, DRP and EAF. One of these aspects is to have a comprehensive, agent-based model that can incorporate the key conditions prevailing at other steelworks in Europe and that will provide directions and potential solutions for the most optimal transition pathway. Therefore, local conditions and product aspects need to be part of a scenario analysis to support the implementation of the agent-based model.

In that respect, the results obtained from the evaluation of the KPIs will serve as input for Task 3.4, Task 3.5 and Task 3.6 of WP 3: Scenario studies to reoptimize process integration in future integrated steelworks. Clearly, this an iterative process where the various results from WP 1 and WP 3 will be linked: output of the scenario analysis will be used as input for WP 1 and updated results of WP 1 will serve as input for WP 3 to elucidate the key ingredients of the transition pathway for integrated steelworks.

With respect to local conditions there are several differences on the country level, both with respect to national aspects and at the process level. Optimization and the best possible solution in terms of maximizing production of steel products per year and minimizing CO_2 emissions is driven by cost. Here, we use five main cost drivers, which to some extent are site specific, including price of electricity, price of renewables (natural gas, hydrogen, biochar), price of scrap, and price of emitting/handling CO_2. Especially the transport of CO_2 will be location-dependent as dedicated infrastructure for CO_2 transport and storage is (yet) not available in most European countries. Several options are, in theory, available for transport, ranging from transport of compressed CO_2 by pipeline to transport of liquified CO_2 by ship, inland barges, train or by truck.

As an example of aspects related to product quality, Tata Steel produces carbon steel, which means that the targeted product of the direct reduction process is DRI (direct reduction iron) with around 4 wt% carbon. This puts a constraint on the gas composition and flow rate of the reducing and cooling gases used in the direct reduction process. The carbon footprint of the direct reduction process depends on the composition of the reducing gas to achieve a given production level and to have the required properties of the DRI, on the one hand, and the source of the (renewable) hydrogen and natural gas, on the other hand. Options for reducing and cooling gas based on the use of renewables include electrolytic hydrogen, bio-based hydrogen, synthetic natural gas, and biomethane. In Figure 5.2 an example is given of a gas-based direct reduction furnace to indicate options for the composition of the reducing gas, in this case 20 vol% green hydrogen (H_2) and 80 vol% of renewable methane (CH_4), and options for renewable natural gas for the cooling gas.

Finally, the energy mix of the electrical power grid will depend on local availability of renewable sources, such as the fraction of onshore and offshore wind, the fraction of solar PV, and access to hydropower. As this energy mix for the power grid will be different for different countries, there will be clear differences in the overall carbon footprint of the different future steelworks throughout Europe. In Table 5.1 a high-level overview is given of the (relative) relevance of the various options for the use of renewables in the four main ironmaking units.

In the following a breakdown is given of the various conditions and requirements that need to be considered in developing a feasible agent-based model to support the transition to and implementation of the DRP - EAF route in both existing and new steelworks. In order to have flexibility in terms of the use of renewables – like intermittent electricity, different compositions for the reducing gas, combining green hydrogen with biomethane, use of biochar in both the BF and the EAF, and a variable amount of scrap – the future configurations of integrated steelworks will have to be able to incorporate variations in feedstock composition in the day-to-day operation.

1. National aspects

- Carbon intensity of the national electrical power grid, specifics of the energy mix for the power grid in terms of relative contribution of different types of renewables and total installed capacity.

2. Process level aspects

- Incorporation of intermittent renewable electricity, including electricity from onshore and offshore wind and solar PV (photovoltaic).
- Gas composition of reducing gas, based on options for using different forms of natural gas (fossil natural gas, methane, synthetic natural gas, biomethane), hydrogen (electrolytic, bio-based) and bio-syngas.
- Use of scrap and biomass (biochar) to reduce carbon footprint, enhance circularity, and to optimize the use of renewable electricity.

3. Operational cost aspects

- Price of electricity
- Price of reducing gas, based on the use of different sources for natural gas and hydrogen
- Price of biochar
- Price of scrap
- Cost of emitted CO_2 (EU ETS) and transport and storage cost of CO_2

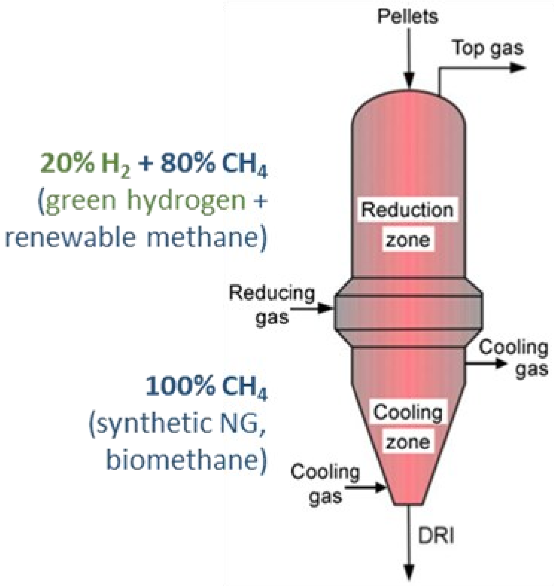

Figure 5.2. Gas-based direct reduction shaft furnace, with as an example feeding of a 20 vol% H_2 + 80 vol% CH_4 reducing gas and the use of synthetic natural gas (NG) or biomethane as the cooling gas.

Table 5.1. Matrix of process parameters versus main iron making units, with blast furnace (BF), basic oxygen furnace, (BOF), gas-based direct reduction process (DRP), and electric arc furnace (EAF).

Process / unit	BF	BOF	DRP	EAF
Energy mix of power grid	0	0	+	++
Reducing gas (CH_4, H_2, syngas)	0	0	++	0
Biomass / biochar	+	0	0	++
Scrap	++	++	0	++

0: Less relevant.

+: Some impact on the reduction of the carbon footprint to be expected.

++: One of the main options to reduce carbon emissions, both in the form of carbon dioxide, CO_2, and in the form of carbon monoxide, CO.

6 Conclusions

After considering different options for realizing the multi-agent systems proposed in the AgiFlex project, the partners have agreed to develop a custom MAS framework to enable the required flexibility and modularity for implementation in different steel plants. A transition to low carbon emission steel production not only entails large changes in the production processes, but also continuous adaptation to evolving circumstances as current energy systems and market conditions are under constant development. Progress in, e.g., renewable energy and hydrogen technologies has shown both rapidity and unpredictability, which together with the site-specific conditions and steel plant transition pathways places key importance to the flexibility and adaptability of the MAS framework.

Besides KPIs measuring the general economic and environmental performance of a steel plant, KPIs capable of capturing site-specific goals such as targeted DRI production rates and use of recycled scrap have also been included. Different types of data required for realizing simulations and KPI evaluations were identified. The work in WP1 will be linked to the digital twin development in WP2 and the scenario studies in WP3, with all WPs contributing to the overall progress of the MAS framework and the forthcoming implementations at the demonstrator sites.

7 References

- Wooldridge, M. (2009). An Introduction to MultiAgent Systems (2nd ed.). John Wiley & Sons.
- Bellifemine, F., Caire, G., & Greenwood, D. (2007). Developing Multi-Agent Systems with JADE. John Wiley & Sons.
- Bordini, R. H., Hübner, J. F., & Wooldridge, M. (2007). Programming Multi-Agent Systems in AgentSpeak using Jason. John Wiley & Sons.
- Masad, D., & Kazil, J. (2015). MESA: An Agent-Based Modeling Framework. Proceedings of the 14th Python in Science Conference (SciPy), 53–60.
- Mariani, S., Picone, M., & Ricci, A. (2022). *About Digital Twins, agents, and multiagent systems: A cross-fertilisation journey*.
- van Dyck, M., Lüttgens, D., Piller, F. T., & Brenk, S. (2023). *Interconnected digital twins and the future of digital manufacturing: Insights from a Delphi study*.