SCHOOL OF
MANAGEMENT AND
TECHNOLOGY
POLYTECHNIC OF
PORTO

MASTER
INFORMATICS ENGINEERING

Reference Architecture for Sustainable Manufacturing-as-a-Service driven by Digital Twin

Diogo Alves Machado Neto

10/**2025**

**SCHOOL OF
MANAGEMENT AND
TECNOLOGY**
POLYTECHNIC OF
PORTO

MASTER
INFORMATICS ENGINEERING

Reference Architecture for Sustainable Manufacturing-as-a-Service driven by Digital Twin

Diogo Alves Machado Neto
8200435

Advisors

Professor Doutor Cristóvão Dinis Polido de Sousa
Professora Doutora Carla Sofia Gonçalves Pereira

Dissertation Report submitted in fulfilment of the requirements for the Master's degree in Informatics Engineering in the School of Management and Technology of the Polytechnic of Porto.

10/**2025**

Integrity Statement

I, Diogo Alves Machado Neto, student nº 8200435, of the Master's Degree in Informatics Engineering of the School of Management and Technology of the Polytechnic of Porto, declare that I have not plagiarized or self-plagiarized, therefore the work entitled ”Reference Architecture for Sustainable Manufacturing-as-a-Service driven by Digital Twin” is original and of my own authorship, not having been used previously for any other purpose. I further declare that all sources used are cited, in the text and in the final bibliography, according to the referencing rules adopted in the institution.

Acknowledgements

I would like to express my deepest gratitude to the individuals and institutions who have supported me throughout this master's thesis journey. This work would not have been possible without their guidance, encouragement, and support.

First and foremost, I extend my sincerest thanks to my thesis advisors, Professor Doctor Cristóvão Sousa and Professor Doctor Carla Pereira. Their invaluable guidance, insightful feedback, and unwavering support have been instrumental in shaping this research. Their expertise and mentorship have not only guided me through the complexities of this topic but have also been crucial in the development of this research.

I am also deeply grateful to ESTG, which has been my academic home for both my Bachelor's and Master's degrees. The institution has provided the foundational knowledge and supportive environment that have shaped my academic journey from the very beginning.

Finally, I would like to thank my family and friends for their constant encouragement, patience, and belief in me. Their support has been a source of strength throughout my studies.

This work has been supported by the European Union under the Next Generation EU, through a grant of the Portuguese Republic's Recovery and Resilience Plan (PRR) Partnership Agreement, within the scope of the project PRODUTECH R3 – ”Agenda Mobilizadora da Fileira das Tecnologias de Produção para a Reindustrialização”, Total project investment: 166.988.013,71 Euros; Total Grant: 97.111.730,27 Euros.

Abstract

The industrial paradigm is shifting towards a data-driven Circular Economy, a transition hindered by a fragmented digital landscape of data silos. Regulatory drivers, such as the European Union's Digital Product Passport, are transforming standardized and sovereign data sharing into a legal necessity, creating an urgent need for a cohesive solution. This thesis addresses the lack of a holistic architectural framework for industrial data sharing by proposing a reference architecture for the Internet of Digital Twins. Following a Design Science Research methodology, this work presents a Reference Architectural Model for Industry 4.0-compliant blueprint that cohesively integrates key standards for data modeling, sovereign data sharing infrastructure, and a decentralized cryptographic trust layer. The architecture's viability is validated through a two-part experimental approach. First, a Proof of Concept demonstrates the technical feasibility of one of the architecture's layers by automating the aggregation and standardization of data from both Operational Technology and Information Technology sources. Second, a scenario-based Proof of Value argues for the utility of the full architecture in enabling complex, sustainable manufacturing applications. The primary contribution of this work is, therefore, a validated and prescriptive blueprint for building the secure, interoperable, and trustworthy data ecosystems required for the data-driven circular economy.

Keywords: Industry 5.0, Digital Twins, Data Spaces, Digital Product Passport, Industrial Data Interoperability, Sustainability

Resumo

O paradigma industrial encontra-se em transição para uma Economia Circular orientada por dados, uma mudança dificultada por um panorama digital fragmentado e caracterizado por silos de dados. Fatores regulamentares, como o futuro Passporte Digital do Produto da União Europeia, estão a transformar a partilha de dados soberana e normalizada numa necessidade legal, criando uma necessidade urgente de uma solução coesa. Esta tese aborda a falta de uma arquitetura holística para a partilha de dados industriais, propondo uma arquitetura de referência para a Internet of Digital Twins. Seguindo a metodologia de Design Science Research, este trabalho apresenta um modelo prático e prescritivo, compatível com o Reference Architectural Model for Industry 4.0, que integra de forma coesa os principais standards para modelação de dados, infraestrutura de partilha de dados soberana e uma camada de confiança criptográfica e descentralizada. A viabilidade da arquitetura é validada através de uma abordagem experimental em duas partes. Primeiramente, uma Proof of Concept demonstra a viabilidade técnica de uma das camadas da arquitetura, automatizando a agregação e normalização de dados de fontes de Operational Technology e Information Technology. Em segundo lugar, uma Proof of Value baseada em cenários argumenta a favor da utilidade da arquitetura completa na viabilização de aplicações complexas de manufatura sustentável. A principal contribuição deste trabalho é, portanto, um modelo validado e prescritivo para a construção dos ecossistemas de dados seguros, interoperáveis e confiáveis necessários para a economia circular orientada por dados.

Palavras-chaves: Indústria 5.0, Digital Twins, Data Spaces, Passaporte Digital do Produto, Interoperabilidade de Dados Industriais, Sustentabilidade

Contents

List of Figures

List of Tables

Listings

Acronyms

3D Three-Dimensional.

AAS Asset Administration Shell.
AI Artificial Intelligence.
API Application Programming Interface.

B2B Business-to-Business.

DID Decentralized Identifier.
DPP Digital Product Passport.
DSR Design Science Research.
DT Digital Twin.

EBSI European Blockchain Services Infrastructure.
eDSR echeloned DSR.
ERP Enterprise Resource Planning.
ESPR Ecodesign for Sustainable Products Regulation.
ETL Extract-Transform-Load.

I4.0 Industry 4.0.
I5.0 Industry 5.0.
IDS International Data Spaces.
IDS-RAM IDS Reference Architecture Model.
IDSA International Data Spaces Association.
IoDT Internet of Digital Twins.
IT Information Technology.

JSON JavaScript Object Notation.

KMS Key Management System.

LCA Life Cycle Analysis.
LCI Life Cycle Inventory.
LLM Large Language Model.

MCP Model Context Protocol.
MES Manufacturing Execution System.
MQTT Message Queuing Telemetry Transport.

OPC UA Open Platform Communications Unified Architecture.
OT Operational Technology.

P2P Peer-to-peer.

RAMI 4.0 Reference Architectural Model for Industry 4.0.

SDK Software Development Kit.
SMaaS Sustainable Manufacturing-as-a-Service.
SMEs Small and Medium Enterprises.

URL Uniform Resource Locator.

VC Verifiable Credential.

W3C World Wide Web Consortium.
WoDT Web of Digital Twins.
WP8 Work Package 8.

Chapter 1

Introduction

For over a decade, the narrative of industrial progress under the banner of Industry 4.0 (I4.0) was one of relentless optimization. The primary goals were efficiency, automation, connectivity, and flexibility, aimed at producing goods faster and at a lower cost [1]. While this paradigm yielded significant technological advancements, a new vision is now taking hold. Often termed Industry 5.0 (I5.0), this emerging paradigm, proposed by the European Commission, complements the I4.0 with a renewed focus on the European industry becoming more sustainable, human-centric, and resilient [2]. At its core is the transition to a Circular Economy, which aims to eliminate waste by promoting the continued use of resources through strategies like repair, reuse, remanufacturing, and recycling [3].

This transition to a circular and sustainable industry is not merely a conceptual shift; it is a profound data challenge. The ability to execute circular strategies depends entirely on the availability of standardized, trustworthy, and interoperable data that can flow seamlessly across entire value chains. However, this vision is hindered by a persistent reality of fragmented data landscapes, both within and between organizations.

The main problem is that industrial data is trapped. Within a single enterprise, real-time data from the factory floor (Operational Technology (OT)), including legacy systems, is often disconnected from the transactional data in enterprise systems (Information Technology (IT)). Between enterprises, valuable information is locked in proprietary formats, preventing the collaboration needed for a resilient and circular value chain.

The issue of data fragmentation is now being driven by regulatory action. Mandates such as the European Union's upcoming Digital Product Passport (DPP) are making data interoperability an unavoidable necessity from both legal and technological perspectives [4]. By requiring a comprehensive digital record for products across their lifespan, the DPP makes solving the industry's data sharing challenge an urgent priority.

For many companies, and especially for Small and Medium Enterprises (SMEs), the technical and organizational hurdles to meet these new requirements are huge. They face a confusing landscape of powerful but disconnected standards and technologies, without a clear, accessible, and holistic architectural blueprint to guide them [5]. The central problem this thesis addresses is the lack of such a guidebook: a comprehensive, integrated architecture for industrial data sharing to ensure upstream and downstream data flow in a continuous and standardized manner, connecting the shop floor to the cloud, and boosting the implementation of circularity-oriented instruments such as the DPP.

To accomplish this, a conceptual overview of the Internet of Digital Twins (IoDT) was developed. IoDT is fully compliant with the Reference Architectural Model for Industry 4.0 (RAMI 4.0) and is designed to serve as a practical guideline for an end-to-end data architecture, providing the foundations to implement a sovereign, interoperable, and sustainable data ecosystem. It is not a new standard but rather a holistic framework that integrates the most important existing standards for data integration, standardization, sharing, and trustability.

The main contribution of this work is designing a reference architecture that could act as a practical blueprint for the industry, especially for SMEs, by providing a clear, layered approach that addresses the entire data lifecycle, from the factory floor to the global value chain. IoDT aims to provide the foundational framework required for participation in the data-driven circular economy.

This thesis is structured to build a clear and logical argument for the proposed architecture. It begins in Chapter 2 with a critical evaluation of the current industrial data landscape, identifying the key challenges and limitations of existing solutions. Building on this analysis, Chapter 3 details the proposed conceptual architecture of the IoDT, explaining the responsibilities of each tier and specifies the core network infrastructure required for decentralized identity management and semantic interlinking. Chapter 4 then outlines the Design Science Research (DSR) methodology that provides the formal framework for the creation and evaluation of the artifacts. Following this methodology, Chapter 5 presents the experimental validation, consisting of a technical Proof of Concept for one of the architecture's tiers and a scenario-based Proof of Value for the holistic architecture. Finally, Chapter 6 summarizes the contributions of this work and outlines directions for future research and developments.

Chapter 2

The Challenge of Interoperable Industrial Data

The successful transition to a sustainable and circular industrial economy depends on a single, critical enabler: the seamless, standardized, and sovereign exchange of data across the entire value chain. While this goal is widely acknowledged, its practical accomplishment presents a series of complex, interconnected challenges.

This chapter presents a critical evaluation of the state of the art, beginning with the vision of a Data-Driven Circular Economy. It then analyzes the fundamental data-centric challenges this new paradigm introduces and evaluates the existing instruments and standards intended to address them.

2.1 Data-Driven Circular Economy

Modern manufacturing is driven by data [6]. The goals of increasing supply chain resilience, enabling new service-based business models, and achieving data transparency required for a circular economy are all fundamentally dependent on one thing: the ability to share the right data, with the right partners, at the right time and by means of a standardized architecture [7]. However, this vision is hindered by a persistent reality of data fragmentation and a lack of interoperability, both within and between organizations.

To implement circular business models, there is a need to deploy several decision points across the value chain of a particular product or service [8]. If, on one hand, company-specific decision points are, in theory, easy to deploy, on the other hand, challenges are faced outside the company boundaries [9]. Additionally, those decision points need to be fed by data. Otherwise, no decisions could be made. In this sense, the circular economy requires a new perspective on data governance across the value chain, encompassing data traceability, data transparency, and data sharing. Therefore, the transition to a circular economy can be described as a Three-Dimensional (3D) Data Challenge.

2.2 The Three-Dimensional Data Challenge

2.2.1 Data Traceability

Traceability is critical for enabling the circular economy. It provides the data trail necessary to verify the history and composition of products, which is essential for enabling circular strategies such as remanufacturing, refurbishment, and reuse [10].

This new paradigm requires the collection of new data about the product, including its operations, the origin of its materials, and its consumption metrics [7]. This, in turn, necessitates the creation of a comprehensive data inventory for the product's entire lifecycle. This inventory must contain not only static product data but also dynamic data about processes, operations, and locations [11].

A huge obstacle to achieving traceability is data integration. The data required for a comprehensive digital asset representation does not come from a single source. It originates from two distinct domains within an organization: OT and IT. The first and most significant barrier is the challenge of extracting and consolidating data from these two heterogeneous worlds.

In the OT landscape, the factory floor is extremely fragmented. It frequently contains a mix of modern and legacy equipment from various manufacturers, each speaking its own proprietary communication protocol (e.g., Modbus, PROFINET, EtherNet/IP) [12]. To solve this, the industry relies on industrial gateways and middleware for protocol translation, with Open Platform Communications Unified Architecture (OPC UA) being an important standard that provides an architecture for machine-to-machine communication [13, 14].

Additionally, in the IT landscape, critical data, such as material specifications, supplier information, and production orders, is often locked within monolithic enterprise systems like Enterprise Resource Planning (ERP) and Manufacturing Execution System (MES) systems. Each of these systems typically has its own proprietary data model and requires custom integration efforts, often through bespoke Application Programming Interfaces (APIs) or complex middleware, to expose its data [15].

The current state of the art treats data integration as two separate problems, with different tools and expertise required for OT and IT systems. This approach creates a complex and resource-intensive integration challenge, posing a significant barrier for SMEs [5]. Furthermore, these solutions are primarily designed to solve the problem of data transport, but not necessarily the problem of semantic interoperability. Even when data is successfully extracted from both an OT sensor and an ERP system, it is not guaranteed that it will be structured in a universally understandable format. This reveals a clear and critical need for a standardized target data model that can serve as a common destination for data consolidated from both the OT and IT worlds.

Furthermore, establishing traceability across the value chain introduces an identification challenge. As assets move between different organizations, they must be identifiable in a way that is globally unique, persistent, and verifiable, yet independent of any single centralized authority or proprietary system. Current identification methods often fail to provide this level of decentralized trust, creating breaks in the digital thread when assets cross organizational boundaries. Resolving this challenge is also necessary to fully achieve data interoperability.

2.2.2 Data Transparency

Transparency allows entities in the value chain to increase their visibility over that chain. This visibility is driven by the market, which requires a common, verifiable basis for trust to enable collaboration, and by the internal need to make informed, sustainable decisions [16].

Entities can increase the transparency of the value chain through data carried by the products. This knowledge is not limited to just immediate upstream and downstream partners, but also to all elements of the chain. This aggregated knowledge, available with the product, allows the consumer to have a view of the entire product lifecycle, enabling them to make more informed decisions. This transparency can be achieved through data that is digitally associated with the product, which aggregates knowledge from all elements of the chain. This allows a consumer, for example, to see a product's lifecycle history, or a recycler to identify its material composition, thereby enabling a more sustainable value chain and the creation of new environmentally friendly products [7].

One of the challenges for achieving transparency is the standardization of the data model. Even if data can be successfully extracted, its value is lost if it remains in ambiguous proprietary formats. This lack of a common data language creates new isolated datasets even after integration, limiting interoperability [17] and preventing stakeholders from having a unified view of the product.

However, mere access to standardized data is insufficient without a mechanism to verify its authenticity. In a circular economy, claims about sustainability (e.g., "recycled content," "carbon footprint") are valuable assets. Relying on traditional certificates or unverifiable digital claims creates a trust gap. There is a critical need for a digital method to cryptographically prove the validity of these claims in a decentralized manner, ensuring that transparency does not come at the cost of trust.

2.2.3 Data Sharing

Data sharing is the essential mechanism for collaboration in a circular economy [18]. Its primary value lies in going beyond simple transactions to provide complementary datasets that are crucial for holistic value chain analysis and support decision-making. While the industry mostly relies on a model of Peer-to-peer (P2P) data exchange based on specific schemas for document transfer [19], this transactional view alone is insufficient to support the visibility required for circularity.

To bridge this gap, data sharing must embrace new requirements such as data sovereignty, security, and trust. This requires a shift towards the creation of distributed data ecosystems governed by clear policies for access and resource sharing, rather than simple bilateral connections. Independently of transactional data exchanges continuing to exist, circularity requires these ecosystem-based principles to ensure that data owners retain full control over their assets while collaborating [20].

The central challenge to this is the lack of a trusted and sovereign sharing framework. This is a problem not just of data exchange, but also of business logic and governance. Without a framework that can enforce usage policies and establish trust, companies are rightly hesitant to share valuable data, which affects collaboration [21]. Overcoming this and the other challenges requires specific instruments capable of standardizing these exchanges and enforcing this new model of trust.

2.3 Instruments for a Data-Driven Circularity

These challenges have led to the development of various instruments and standards, such as the DPP for data transparency and data spaces for sovereign data sharing.

2.3.1 Digital Product Passport

The data transparency challenge has been brought into focus by regulatory drivers like the European Union's DPP. The DPP will require a comprehensive digital record for a wide range of products, making transparent data sharing a legal requirement, not just a competitive advantage [4]. By requiring detailed information on everything from material composition to repairability, the DPP acts as a powerful catalyst, forcing the industry to confront its systemic data-sharing problems [10].

While the specific data requirements will vary by product category (with batteries, textiles, and electronics among the first priorities), the Ecodesign for Sustainable Products Regulation (ESPR) outlines a common set of information that a DPP must contain [4]. This typically includes, but is not limited to:

- **Identification Data:** Manufacturer details, product model, unique identifiers, and date of manufacture.
- **Material Composition:** A detailed breakdown of materials and chemical substances used
- **Circularity and Sustainability Data:** Information on recycled content, environmental footprint, CO_2 emissions, etc.
- **Repair and Maintenance Data:** Availability of spare parts, repair manuals, disassembly instructions, etc.
- **Ownership and History:** Records of previous ownership, history of major repairs, etc
- **End-of-Life Information:** Recycling procedures, etc.

Compiling and maintaining this comprehensive set of data throughout a product's life requires a high level of collaboration and data integration among all entities in the value chain, from raw material suppliers to manufacturers, consumers, and recyclers. This makes a decentralized infrastructure for data sharing a necessity to ensure that each entity has full sovereignty over its data.

2.3.2 Data Spaces

A data space is a decentralized ecosystem of peers who agree on a common set of principles and standards for sharing data in a sovereign manner [20]. These principles ensure a fair and value-creating environment, with data sovereignty (the principle that data owners must retain full control over their data at all times) being the most fundamental concept. This is supported by Trust, established through transparent governance and verifiable identities, and Interoperability, achieved through common technical and semantic standards.

To implement these principles, the International Data Spaces Association (IDSA) has developed the IDS Reference Architecture Model (IDS-RAM), which provides a standardized blueprint for building data spaces. The latest version, IDS-RAM 4.0, presents the architecture from five distinct perspectives: a Business Layer, a Functional Layer, a Process Layer, an Information Layer, and a System Layer [22].

At the heart of the architecture are the participants and the software components they use to interact. The primary component for any participant is the IDS Connector, a secure software gateway that acts as the sole technical endpoint for all data exchanges. For the data space to function, a set of Essential Services must be in place, including an Identity Provider for authentication, a Metadata Broker for discovery, and a Dynamic Attribute Provisioning Service for authorization. Optional Parity Services like a Clearing House can provide auditable logs of data transactions. Figure 2.1 provides a high-level summary of the interactions between these components, as defined in the Process Layer of the IDS-RAM. It is important to note that, as the official documentation clarifies, the Identity Provider is omitted from this specific diagram to maintain readability.

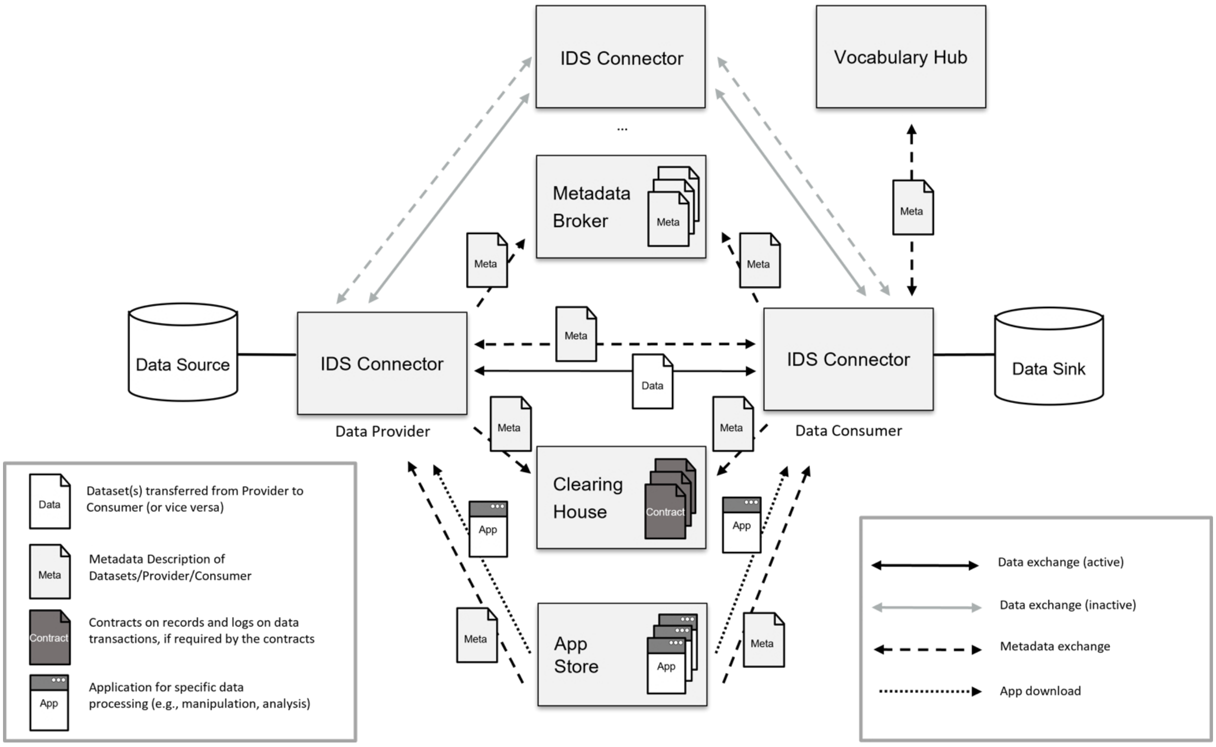


Figure 2.1: Overview of interactions between IDS Components (Process Layer)
Source: IDS-RAM 4.0 [22]

The IDS architecture proves to be a great solution for data sharing, as demonstrated by large-scale industrial initiatives like Catena-X for the automotive industry [23]. Catena-X has successfully created a value chain data space for applications like carbon footprint tracking and parts traceability. However, from the perspective of a universally applicable solution, this approach has limitations. Firstly, these initiatives are often highly domain-specific, with governance rules and semantic models that are specific to a single industry.

This makes them less accessible for SMEs in other sectors who may lack the resources to adapt to these ecosystems built for a single purpose [5]. Secondly, while the IDS architecture establishes trust between authenticated participants, it does not provide a native, decentralized mechanism for verifying the authenticity and integrity of the data within or for validating specific claims, such as sustainability certifications.

2.4 Standardizing the Digital Representation of Assets

The data required for circularity goes beyond simple transactional data. A data model that can encapsulate all this information to be transmitted through the data spaces. This is the role of the Digital Twin (DT).

2.4.1 Digital Twin

The DT, though popularized within the I4.0 framework, can be traced back to NASA's Apollo program, where identical physical simulators were built on the ground to mirror the vehicle in space, allowing engineers to diagnose and solve problems remotely [24]. In its modern version, a DT is a dynamic virtual representation of a physical asset, process, or system.

According to Kritzinger et al., a fully realized DT is not a single model but a complex system comprising three essential parts: the physical entity in the real world, its virtual counterpart in the digital world, and the integrated data connection that links them [11]. The richness and directionality of this connection determine the maturity of the implementation, which can be understood as an evolutionary progression.

The most basic stage is the Digital Model, a digital version of an asset without automated data exchange, such as a static 3D CAD model that is updated manually. The next level is the Digital Shadow, where a one-way automated data flow is established from the physical asset to the digital object. This allows for real-time monitoring as sensors update the digital "shadow," though the flow is not reciprocal. The highest level of integration is the Digital Twin, characterized by a bi-directional, automatic data flow. Here, the twin not only mirrors the physical asset but can also influence it, creating a closed loop where simulations and optimizations are fed back as control commands or operational adjustments. [11]

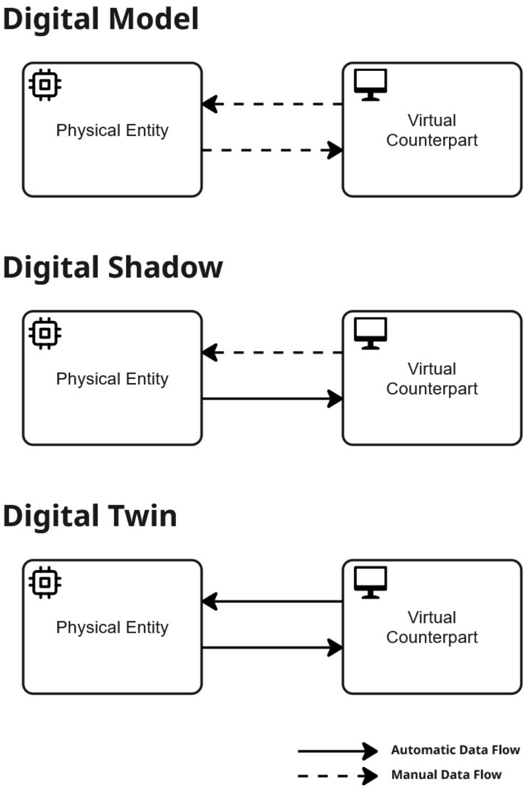


Figure 2.2: Maturity levels of Digital Twin implementations.
Adapted from Kritzinger et al. [11]

The DT's applications in the manufacturing world are widespread across the value chain. It begins in the design and engineering phase with virtual prototyping and simulation to optimize products before physical creation and extends into manufacturing and commissioning, where virtual commissioning reduces ramp-up time and errors [25, 26]. Throughout the operation and service phase, the DT enables advanced prognostics and health management systems for predictive maintenance [27], famously used in assets like wind turbines to predict component lifespan and prevent failures [28]. Finally, the DT can also be used as a means to store a product's information across its entire lifecycle, providing a complete record of material composition and wear characteristics, which is invaluable for disassembly, remanufacturing, and recycling, thereby directly supporting the creation of a DPP [29, 30].

Despite this immense potential, a significant historical challenge to the widespread adoption of DTs has been the lack of standardization. Implementations have often been domain-specific and proprietary, creating data silos and hindering the interoperability required for a connected, cross-company value chain. [17]

2.4.2 Asset Adminsitration Shell as a Standardized Implementation of a Digital Twin

In 2016, the German-led Plattform Industrie 4.0 introduced the Asset Administration Shell (AAS) as a core component of the RAMI 4.0. The AAS provides a standardized digital representation of an asset, which can be anything from a single motor to a complete production line, a software component, or even a human operator [31].

The AAS is not the Digital Twin itself, but rather the standardized implementation of it. It acts as a digital container that organizes all information and functionalities related to an asset throughout its lifecycle. To achieve interoperability, the AAS specification [32] defines a detailed meta-model, the key components of which are the Shell, Asset Information, Submodels, and the semantic referencing system.

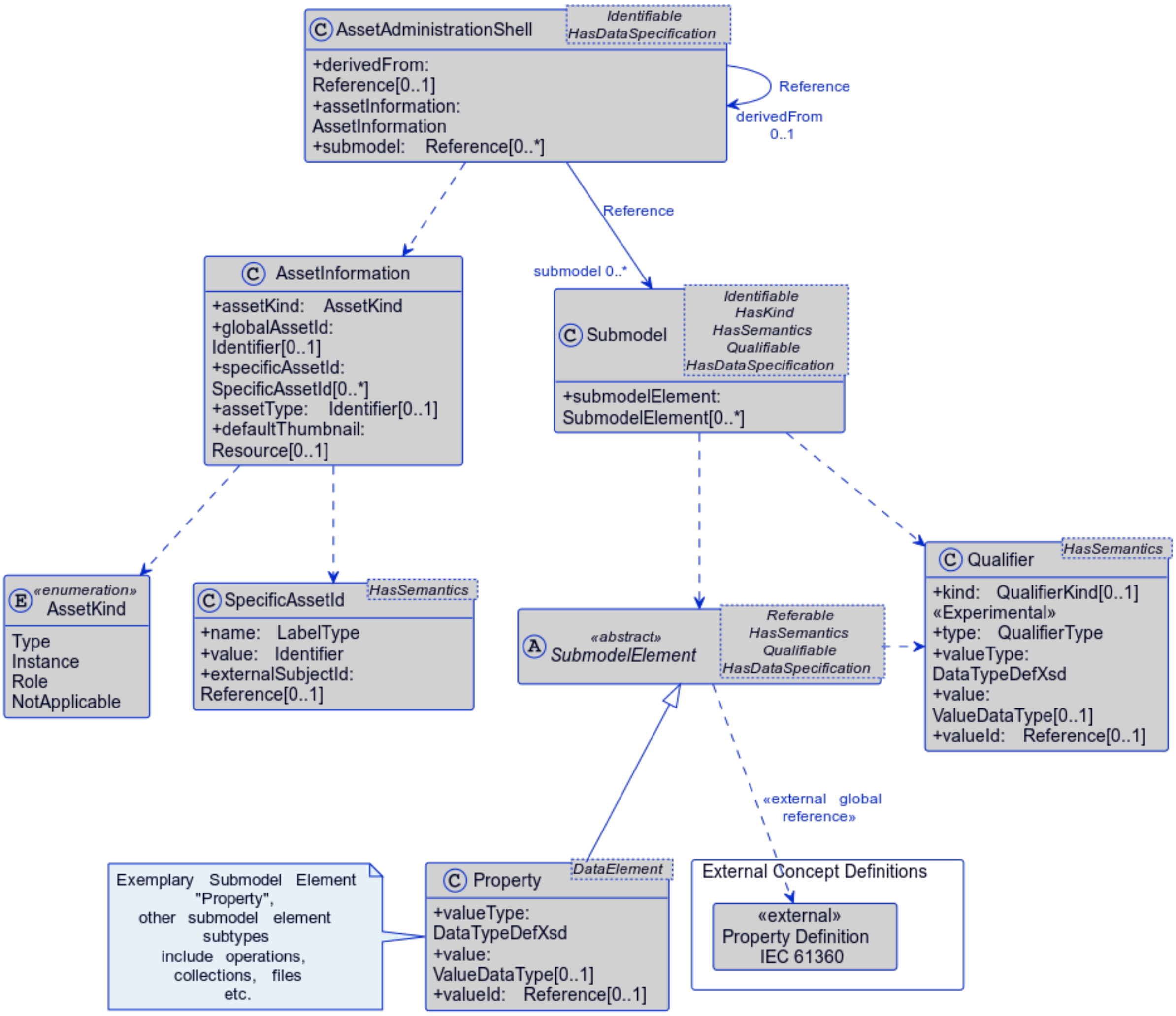


Figure 2.3: Overview Metamodel of the Asset Administration Shell
Source: Industrial Digital Twin Association (IDTA) [32]

The AAS itself is the top-level component, acting as the main entry point. Each AAS is uniquely identifiable via a globally unique `Identifier` and contains descriptive metadata. Crucially, it does not hold all the data directly, but rather contains one or more references to Submodels, which can be hosted independently.

The Shell is unambiguously linked to the physical world via its `AssetInformation` component. This component holds critical identifiers for the physical asset, including a `globalAssetId` for worldwide uniqueness and optional `specificAssetId` pairs (e.g., a serial number) for more granular identification. It also defines the `AssetKind` (e.g., 'Type' or 'Instance') to specify whether the AAS describes a product model or a specific physical object.

The core of the AAS's content is held within Submodels. A Submodel is a structured collection of data and functions that represents a specific aspect or facet of the asset. This modular approach is highly flexible; for example, an electric motor might have a Submodel for its technical specifications, another for its operational history, and a third for sustainability metrics. This separation of concerns allows different stakeholders to interact with only the information relevant to them.

Each Submodel is composed of a collection of Submodel Elements, which are the actual data points. The specification defines an abstract `SubmodelElement` from which various concrete types are derived to represent diverse information. A few examples of these types are:

- `Property`: A simple key-value pair (e.g., ManufacturerName: "Brand Name").
- `File`: A reference to an external document, such as a PDF manual.
- `RelationshipElement`: Defines a directed relationship between the asset and another entity.
- `Operation`: Represents a service that can be invoked on the asset.
- `SubmodelElementCollection`: A kind of struct (i.e. logical encapsulation of multiple named values) with a fixed number of other submodel elements
- `SubmodelElementList`: An ordered list of submodel elements

To add further context, Submodels and Submodel Elements can be annotated with Qualifiers. These are constraints or additional pieces of information that qualify the parent element, such as defining the measurement uncertainty of a sensor reading (`Property`) or specifying the nature of a connection (`RelationshipElement`).

To ensure that these elements are understood unambiguously across different systems, the AAS heavily relies on semantic referencing. Elements that are designated as `HasSemantics` can be linked to an external definition. This is what ensures that a property, regardless of its name, is universally understood.

Much of the related work identifies the AAS as the practical and standardized implementation of the DT [31, 29, 23, 33, 34, 30]. However, the AAS specification itself does not solve the foundational integration problem. It defines the destination but not the entire journey. The challenge of efficiently extracting, transforming, and mapping the heterogeneous data from both OT and IT systems into the standardized structure of the AAS remains a significant undertaking, often requiring the same resource-intensive custom projects that the standard was meant to simplify [35]. A solution that would more tightly couple the integration process with the target AAS model should be explored.

2.5 The Lack of a Holistic Architecture

All of the gaps presented thus far point towards a need for a holistic architecture that incorporates all of these state-of-the-art components in order to achieve the ultimate goal of data interoperability. While individual standards exist for data integration, data standardization, and data sharing, the state of the art largely treats them as separate components that must be manually integrated into complex projects [23, 33, 34]. Some conceptual architectures have been proposed to address this, with the Web of Digital Twins (WoDT) being a prominent example.

The WoDT is a conceptual paradigm that envisions a broad, open, and distributed ecosystem of interconnected DTs, inspired by the principles of the World Wide Web. In this view, DTs are not seen as standalone applications, but as components of a service-oriented software layer that virtualizes a physical reality. The core of the WoDT concept is the focus on the dynamic relationships between DTs. These relationships are meant to be explicitly captured and represented as links, similar to hyperlinks on the web. To give these links meaning, the WoDT leverages Semantic Web technologies, representing the entire ecosystem as a Distributed Knowledge Graph, where individual DTs are described using languages like RDF and OWL. This allows for rich, machine-readable descriptions of how assets are interrelated, forming a "web" of digital counterparts that can be navigated and queried. The WoDT is therefore designed to be a flexible and dynamic virtual environment, particularly suited for smart applications and multi-agent systems that can exploit this graph of relationships to reason about and interact with the physical world. [36]

But while the WoDT offers a promising conceptual model for an interconnected world of DTs, it also reveals its limitations as a direct solution for the gaps identified throughout this chapter. These gaps stem from the inherent openness that defines the WoDT:

- The WoDT is intentionally abstract and not bound to a specific data model. While it suggests the use of Semantic Web technologies, it does not mandate a standardized and semantically rich meta-model for the DT themselves. This flexibility leaves the door open for the same interoperability problems that the AAS was specifically created to solve.
- The nature of the WoDT lacks a formal framework for data sovereignty and usage control. In industrial value chains, the ability to enforce strict policies on who can access what data and for what purpose is a non-negotiable requirement. The WoDT open approach does not inherently include the governance mechanisms provided by an architecture like the IDS.
- The WoDT concept does not include a native, decentralized trust layer. There is no discussion of how to cryptographically verify the identity of a DT or the authenticity of the claims it makes (e.g., a sustainability certificate) in a way that is independent of a central authority.

In essence, while the WoDT provides a compelling vision for a linked and intelligent digital world, it lacks the concrete, prescriptive components for standardization, sovereignty, and trust that are essential for building an interoperable industrial data ecosystem. This leaves a clear and significant gap for a more holistic architecture, in which the DT goes far beyond the mere digital representation of a physical asset: the IoDT.

Chapter 3

The Internet of Digital Twins Architecture

This chapter discusses the concept of the IoDT and its alignment with existing reference architectures. The concept of IoDT leverages standards such as the AAS and World Wide Web Consortium (W3C) Decentralized Identifiers (DIDs) to build a network of shareable and verifiable DTs. The IoDT enables a common interaction with DTs, promoting the development of emerging instruments like the DPP.

The IoDT adds a trust layer and is fully compliant with the RAMI 4.0. The IoDT reference architecture was designed to serve as a comprehensive and accessible guideline for industries, particularly SMEs, to build interoperable and sovereign data ecosystems. By integrating solutions for data integration, standardization, and trust into a layered framework, the IoDT provides a practical blueprint that addresses the entire data lifecycle.

3.1 Reference Architectural Model for Industry 4.0

Validating a new industrial architecture requires positioning it within a standardized framework like the RAMI 4.0. Developed by Germany's Plattform Industrie 4.0, it is a three-dimensional model designed to provide a common language and a structured map for discussing and developing Industry 4.0 solutions [37].

The model, as represented in Figure 3.1, consists of three axes:

- **Hierarchy Levels:** This axis represents the different functional levels within a factory and its wider network, ranging from the individual "Product" and "Field Device" on the shop floor, up through the "Control Device," "Station," "Work Center," "Enterprise," and finally to the "Connected World."
- **Life Cycle Value Stream:** This axis describes the entire lifecycle of an asset or product, from its initial development and creation as a "Type" to its real-world deployment and maintenance as an "Instance."
- **Layers:** This axis decomposes the representation of an asset into six distinct layers, representing the transition from the physical world to the business process:

- **Asset Layer:** Represents reality, such as the physical components and machines on the shop floor.
- **Integration Layer:** Handles the transition from the physical to the digital world (e.g., sensors).
- **Communication Layer:** Standardized access to information and protocols.
- **Information Layer:** Manages the necessary data (e.g., the DT data).
- **Functional Layer:** Contains the formal logical functions of the asset.
- **Business Layer:** Orchestrates the organization and business processes.

By providing this three-dimensional map, RAMI 4.0 allows for any component, standard, or use case in I4.0 to be precisely located and understood in its full context. It ensures that all relevant aspects of an architecture are considered, making it an indispensable tool for designing and validating holistic systems like the IoDT.

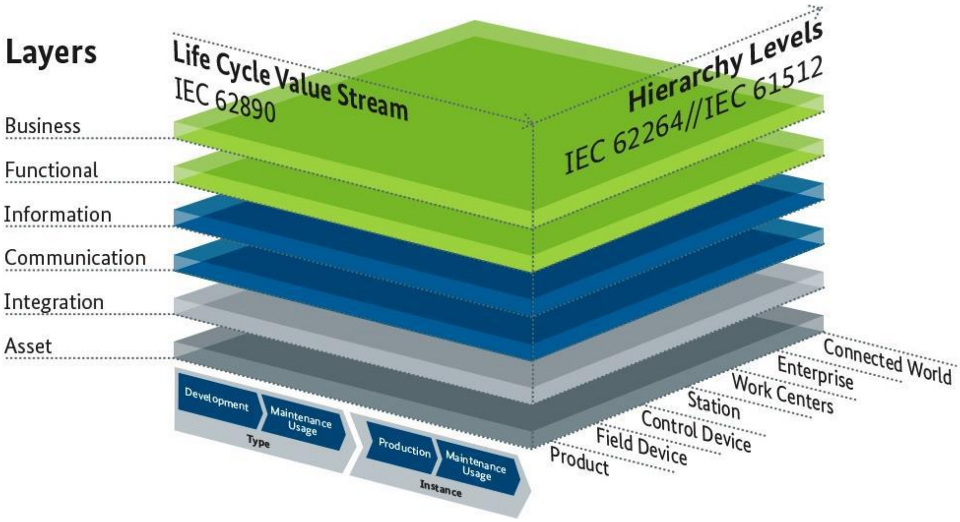


Figure 3.1: Reference Architectural Model for Industry 4.0
Source: Plattform Industrie 4.0 [37]

3.2 A Framework for Trust

In a decentralized ecosystem like the IoDT, where data is exchanged between partners who may not have a pre-existing relationship, a trust layer is necessary. Establishing trust in an enterprise environment requires a framework that combines two important pillars: Governance and Technology.

Governance is the primary motivation for developing a reference architecture (the IoDT) designed to promote transparency. It provides the legal and operational framework that defines how participants agree on usage control, security, and identity. In this context, governance is grounded in the principles of the IDS, ensuring that entities are authorized to participate and that their actions contribute to a transparent ecosystem.

Technology provides the means of verification through cryptographic tools to prove data integrity and authenticity without a central authority. To verify data rather than just participants, the IoDT integrates two W3C standards: DIDs and Verifiable Credentials (VCs).

3.2.1 Decentralized Identifiers

DIDs are a new type of identifier that enables verifiable, decentralized digital identity [38]. Unlike traditional identifiers like domain names or email addresses, which are controlled by centralized registries, DIDs are generated and controlled by the entity they identify (the DID Subject). The core components of the DIDs standard are the identifiers themselves, the DID Documents they resolve to, and the underlying data registry.

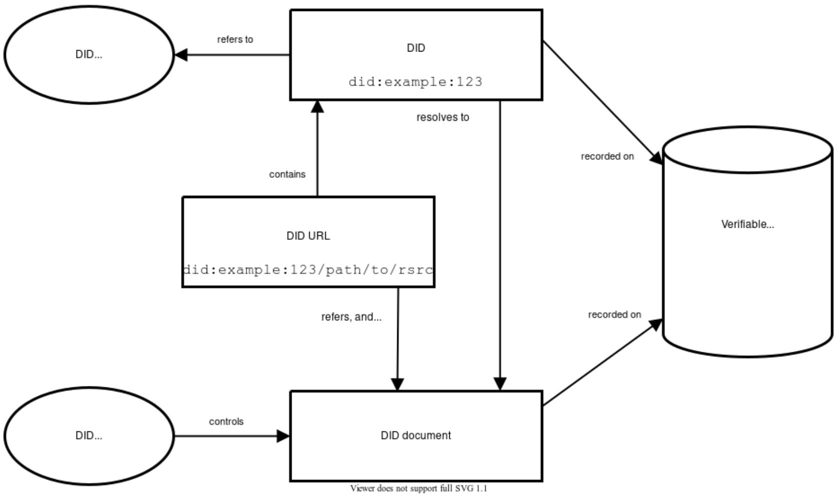


Figure 3.2: Overview of DID architecture
Source: W3C DIDs v1.0 [38]

A DID is a simple string consisting of three parts: the `did` prefix, the identifier for the specific DID Method (which defines the technical implementation), and a unique, method-specific identifier string. The DID method specifies how the DID is anchored in a Verifiable Data Registry, which is any such system that supports recording DIDs and returning data necessary to produce DID documents (e.g., distributed ledgers, decentralized file systems, databases of any kind, peer-to-peer networks).

The process of looking up a DID is called DID Resolution. When a DID is resolved, it returns a DID Document, a JavaScript Object Notation (JSON) file containing the public cryptographic material and service endpoints associated with the DID Subject. This document contains:

- **Verification Methods:** A list of public keys that can be used to verify cryptographic proofs
- **Service Endpoints:** A set of network addresses where the DID Subject can be interacted with

This mechanism allows any entity to cryptographically verify that they are interacting with the legitimate controller of the DID, providing a secure and decentralized way to establish authenticity. By assigning a DID to every AAS, the IoDT ensures that each DT has a universally verifiable identity throughout its entire lifecycle.

3.2.2 Verifiable Credentials

While DIDs provide proof of identity, VCs provide proof of claims or attributes. The VC data model is based on an information flow between three distinct roles, as illustrated in the figure below: the Issuer, the Holder, and the Verifier. [39]

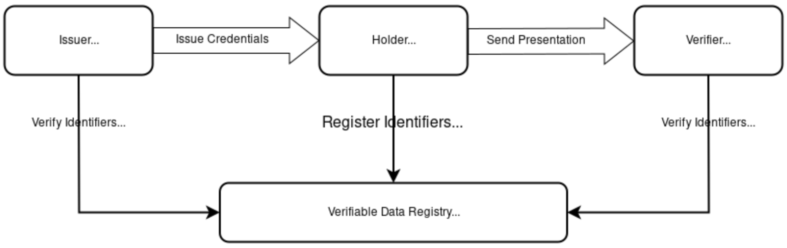


Figure 3.3: The roles and information flows in the Verifiable Credentials ecosystem
Source: W3C VC Data Model [39]

A VC is a set of tamper-proof claims that a trusted entity (an Issuer) makes about a certain subject. The process begins when the Issuer, who holds a DID, cryptographically signs a set of claims (e.g., a product's sustainability data) and issues this digital document to the subject, who becomes the Holder. The Holder stores the VC and can present it to any third party (a Verifier) when they need to prove the claim. The Verifier can then independently and cryptographically check the signature on the VC. To do this, they resolve the Issuer's DID to retrieve their DID Document, obtain the Issuer's public key, and use it to confirm that the signature is valid and the claims have not been altered.

These two standards can be used to create a strong trust layer in the IoDT. In this context, a trusted certification body (the Issuer) can provide a manufacturer (the Holder) with a VC for a low-carbon footprint. The manufacturer attaches this VC to the product's AAS. Any partner or consumer (the Verifier) can then validate this sustainability claim without needing to contact the original certification body.

3.3 IoDT Reference Architecture

The development of the IoDT Reference Architecture is motivated by the need for a comprehensive blueprint to bridge the gap between the fragmented reality of current industrial data landscapes and the interoperability requirements of a data-driven Circular Economy. Its purpose is to establish the necessary trust and transparency required for cross-organizational collaboration while strictly adhering to established standards that ensure full data interoperability. This architecture is structured into three distinct tiers: the Data Tier, the DT Tier, and the APP Tier. This layered approach, compliant with the RAMI 4.0 model, provides a logical progression from raw data acquisition to high-level application deployment.

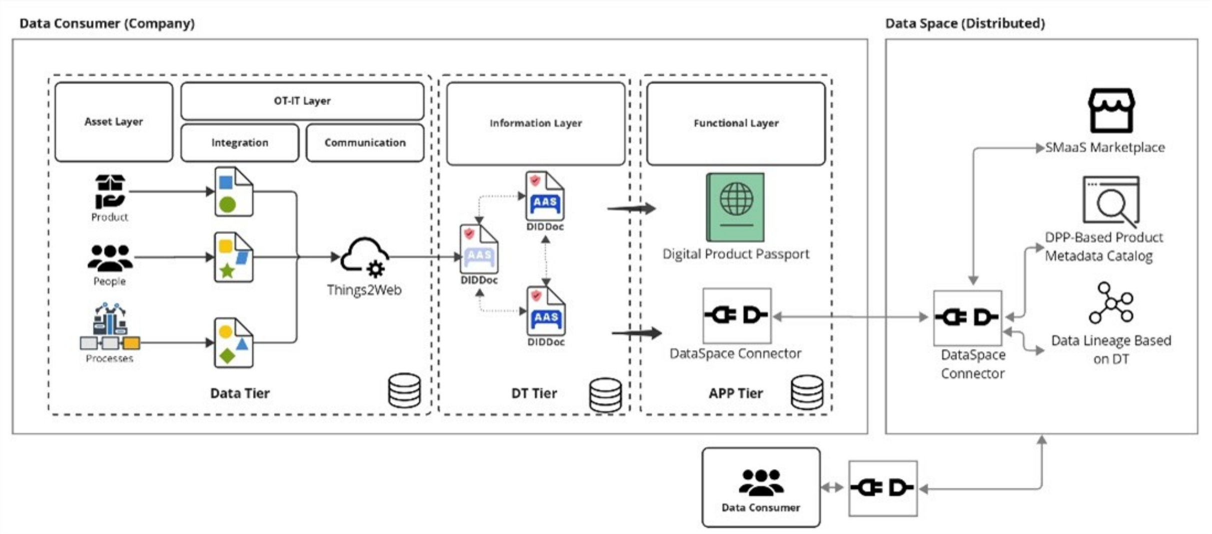


Figure 3.4: Internet of Digital Twins Reference Architecture

The Data Tier is meant to address the Data Integration challenge from both OT and IT. Its primary responsibility is to extract and consolidate data from a diverse range of organizational assets, including physical machinery, enterprise systems (ERP, MES), and even people. This data is then transformed into an AAS and stored in a repository.

The DT Tier builds upon the AAS. Its function is to embed identity and trust directly into the data, creating an interconnected and trustworthy network. This is achieved by assigning a DID to each AAS for a globally unique identity and attaching specific claims to the AAS in the form of VCs. The technical specifications and infrastructure components required to manage these identities are detailed in Section 3.4.

The APP Tier is the final layer where the business logic and value-added services are built upon the secure and interoperable network of Digital Twins. This is where applications such as the DPP are realized. According to our view, a product's DPP is formed by extracting the necessary information from all the AASs related to that product, including the specific shell of that product and the shells of the materials used for its production. The data from the AAS may be made accessible to the wider ecosystem, managed through data space connectors operating within a governed framework like the IDS. But for IDS to be able to share DTs, specific applications should be developed to generate DT-ready artifacts. Since the AAS acts as our DT, AAS-oriented connectors, such as the Eclipse Dataspace Connector[1], may be used.

The opportunity to establish a DT-oriented data ecosystem facilitates the creation of advanced marketplaces that support the Sustainable Manufacturing-as-a-Service (SMaaS) paradigm, specifically endowed with sustainability features [40]. This is achieved by using IDS and DTs with well-defined vocabularies to create a marketplace where services are clearly described and interoperable. The proposed IoDT architecture promotes the development of these sustainability-oriented instruments at both the enterprise and value-chain levels by ensuring a rich dataset that allows users to register and discover products and services based on specific sustainability criteria. Furthermore, by following a bottom-up approach, the IoDT elevates traceability to a new level of maturity. This granular

---

[1] https://eclipse-edc.github.io/

detail contributes to the development of a Life Cycle Inventory (LCI), which is crucial for conducting Life Cycle Analysis (LCA) to determine the eco-friendliness of a product throughout its journey in the value chain. Ultimately, the IoDT functions as a network of well-defined data, promoting value chain visibility, trust, and the opportunity to support sustainability-oriented decisions.

The compliance of the IoDT architecture with RAMI 4.0 can be clearly demonstrated by mapping its tiers across the three axes. Along the Layers axis, the Data Tier corresponds directly to the Asset, Integration, and Communication layers; the DT Tier implements the Information layer by providing a standardized data model; and the APP Tier encompasses the Functional and Business layers where value-added services are realized. The architecture spans the full Hierarchy Levels axis, gathering data from the 'Product' and 'Field Device' levels, processing it at the 'Enterprise' level, and sharing it in the 'Connected World'. Finally, it addresses the entire Life Cycle Value Stream axis by managing an asset's data from its initial design as a 'Type' to its operational deployment as an 'Instance'.

3.4 Structuring the IoDT Core Network

The realization of a true "Internet" of Digital Twins depends on the ability to establish a cohesive network where digital twins are not only isolated artifacts but interconnected entities. This interconnectivity needs robust mechanisms for universally and verifiably identifying assets across different systems and organizations. As mentioned in the previous section, this is the goal of the DT Tier. This section proposes two main components required to support it: an architecture for decentralized identity management and a meta-model for semantically interlinking different assets.

3.4.1 Decentralized Identity Management

A universal system of identity is a fundamental requirement for transforming isolated DTs into a globally interconnected network. However, in this open ecosystem, there's not only the challenge of uniquely identifying an asset throughout the entire value chain, but also verifying its authenticity to ensure it is the genuine article and not a counterfeit. DIDs provide this capability by anchoring the DT's identity to cryptographic proofs. This allows any participant in the value chain to independently verify that a DT actually belongs to the entity it claims to represent. To realize this, the IoDT needs an infrastructure capable of managing these identities and their cryptographic keys.

To realize these capabilities, a technical architecture is proposed for the DT Tier, which manages the lifecycle of DIDs. Figure 3.5 demonstrates this proposed architecture with two workflows: the flow for automated attribution of DIDs upon the creation of an AAS, and another flow demonstrating an example of how the identity of the AAS can be verified.

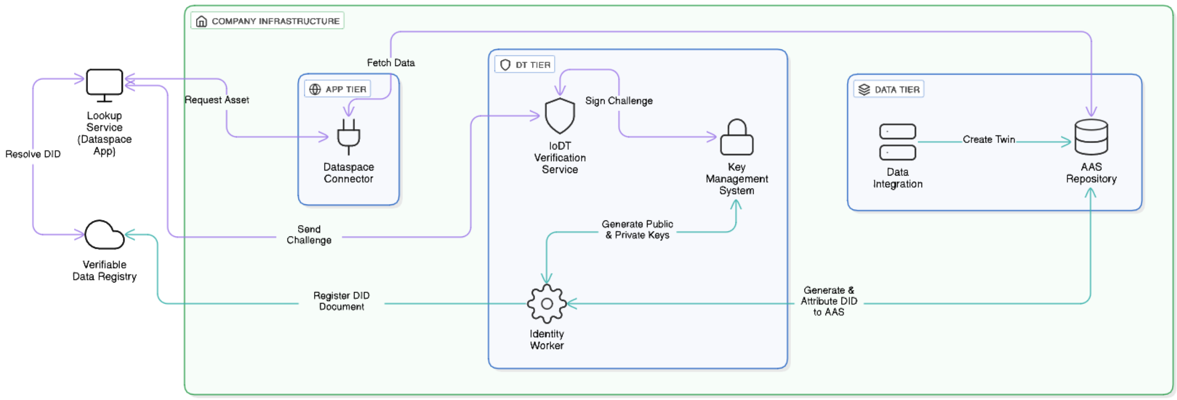


Figure 3.5: DID Attribution and Verification Lifecycles

The company's infrastructure includes the IoDT tiers already introduced in section 3.3. The Data Tier handles the persistence and integration of asset data, comprising the Data Integration layer and the AAS Repository. The APP Tier is represented by the Dataspace Connector, which acts as the sovereign gateway for external data exchange. The main focus, however, is the DT Tier and how it enables the global DT network of the IoDT through DIDs attribution. It comprises three components:

- **Identity Worker:** An orchestration service responsible for the complete lifecycle management of DIDs. It listens for asset creation events and automatically coordinates the generation of cryptographic keys and the registration of DIDs Documents on a public registry.
- **Key Management System (KMS):** A secure system dedicated to storing private keys. It ensures that the DID Document's private key never leaves the trusted boundary of the infrastructure, performing signing operations internally upon request.
- **IoDT Verification Service:** This is an independently exposed API service provided by the company. It serves as the dedicated security endpoint responsible for handling cryptographic challenges. While the Dataspace Connector facilitates the transfer of the AAS data, the Verification Service proves the integrity of that data by interacting with the KMS to sign verification challenges.

Beyond the company's internal boundaries, the architecture relies on shared public infrastructure. The Lookup Service acts as the consumer-facing application (e.g., a DPP viewer) that executes verification checks to guarantee to the user that the presented data is genuine and originates from a verified source. To resolve identities, it interacts with a Verifiable Data Registry which stores the public DID Documents. One example of such a registry is the European Blockchain Services Infrastructure (EBSI)[2].

To support the separation of data access and identity verification, the DID Document must provide specific pointers to the relevant services. The proposed structure utilizes the W3C DID Core specification's service property to define two distinct endpoints: one for retrieving the AAS (via the Dataspace Connector) and another for verifying its authenticity (via the IoDT Verification Service).

---

[2] https://ebsi.eu

The following JSON example illustrates the required structure for an IoDT-compliant DID Document:

```
{
  ”@context”: [
    ”https://www.w3.org/ns/did/v1”,
    ”https://w3id.org/security/suites/jws−2020/1”
  ],
  ”id”: ”did:ebsi:product−12345”,
  ”verificationMethod”: [
    {
      ”id”: ”did:ebsi:product−12345#key−1”,
      ”type”: ”JsonWebKey2020”,
      ”controller”: ”did:ebsi:product−12345”,
      ”publicKeyJwk”: {
        ”kty”: ”EC”,
        ”crv”: ”secp256k1”,
        ”x”: ”dWCvM4fTdeM0KmcaPjDAu6yCkgmrSO0Z5khVdLg97W8”,
        ”y”: ”3123dWCvM4fTdeM0KmcaPjDAu6yCkgmrSO0Z5khVdLg”
      }
    }
  ],
  ”authentication”: [
    ”did:ebsi:product−12345#key−1”
  ],
  ”service”: [
    {
      ”id”: ”did:ebsi:product−12345#aas−access”,
      ”type”: ”AssetAdministrationShellService”,
      ”serviceEndpoint”: ”https://connector.factory−x.com/api/v1/data/aas/product−12345”
    },
    {
      ”id”: ”did:ebsi:product−12345#verification−api”,
      ”type”: ”IoDTVerificationService”,
      ”serviceEndpoint”: ”https://api.factory−x.com/iodt/verify/product−12345”
    }
  ]
}
```

Listing 3.1: DID Document JSON example

DID Attribution Flow

The diagram in Figure 3.6, demonstrates the sequence for obtaining a fully interoperable DT, with verifiable identity, in the IoDT.

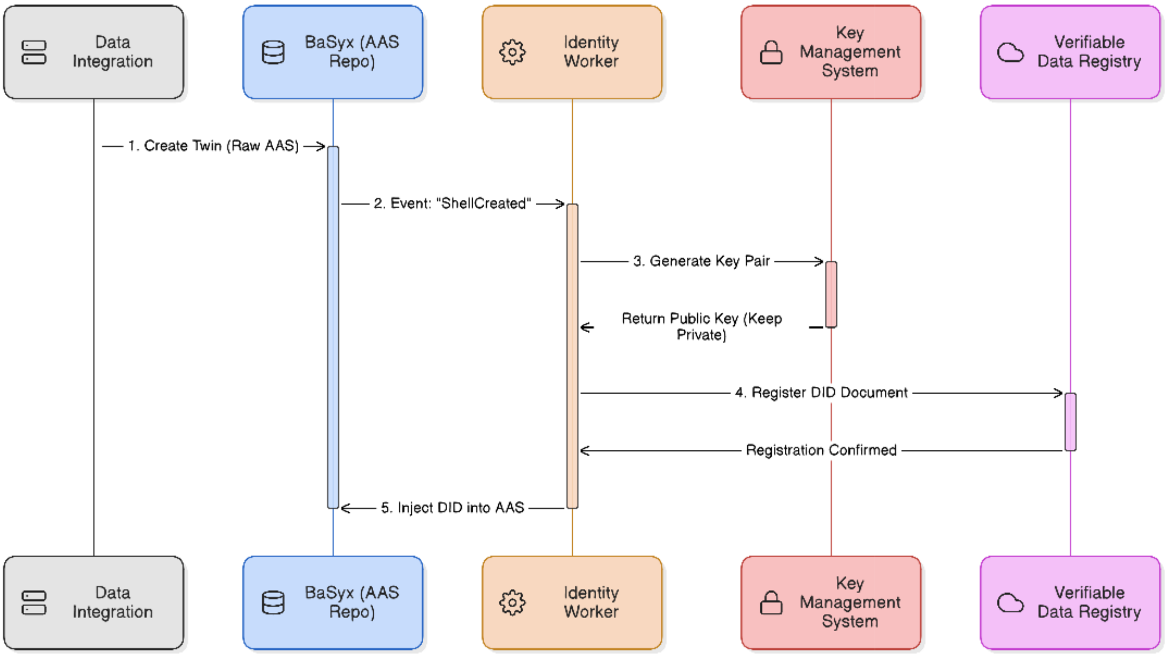


Figure 3.6: Sequence Diagram for DID Attribution

1. **Create Twin:** An AAS is created in the AAS Repository.
2. **AAS Creation Event:** The creation of the AAS triggers an event. The Identity Worker, monitoring these events, initiates the DID generation.
3. **Generate Key Pair:** The Identity Worker first commands the KMS to generate a cryptographically secure key pair (public and private) specifically for this asset. The KMS generates the keys inside its secure enclave. The private key is then permanently stored within the boundary of the KMS and is never exposed. Only the public key is returned to the Identity Worker.
4. **Register DID Document:** The Identity Worker constructs a DID Document containing the Public Key and registers it on the external Verifiable Data Registry.
5. **Inject DID into AAS:** Once the registration is confirmed, the Worker performs a final update to the AAS in the Repository, injecting the DID into the `globalAssetId` field of the `AssetInformation`. This action completes the workflow

DID Verification Flow

Figure 3.7 illustrates an example flow of how an external Lookup Service interacts with the IoDT to retrieve a verified asset.

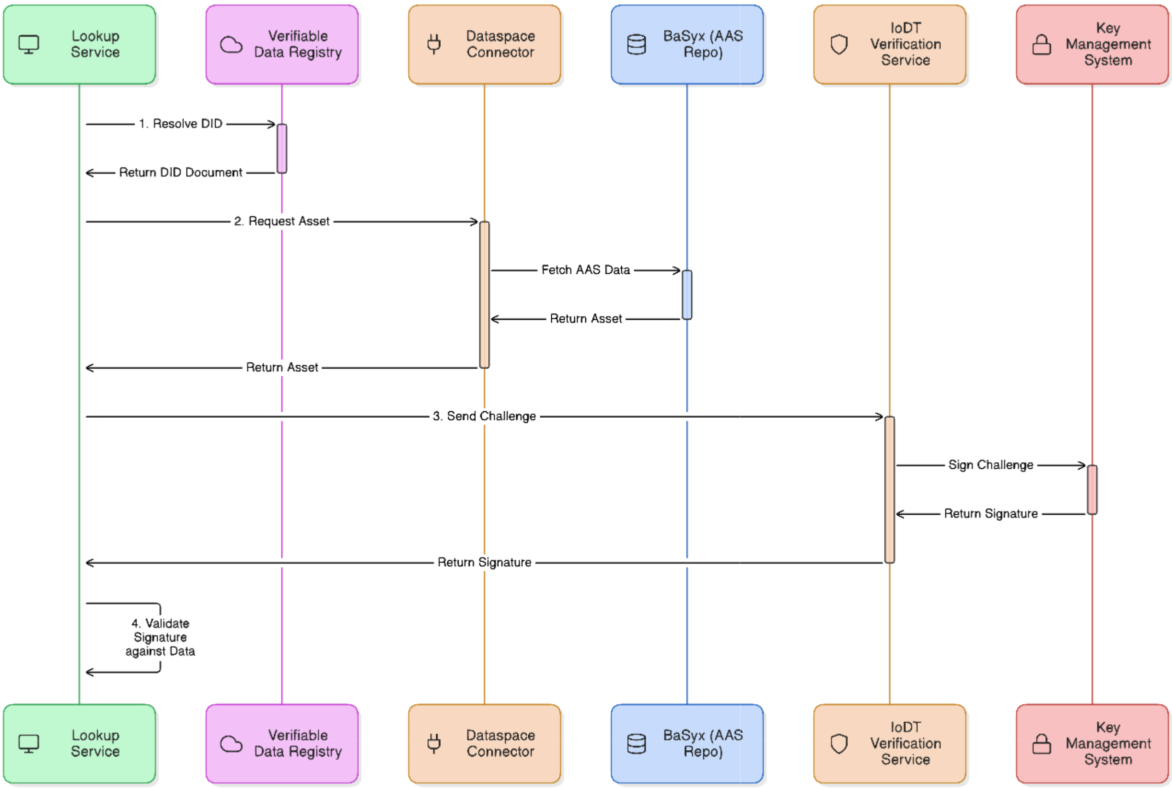


Figure 3.7: Sequence Diagram for DID Verification

1. **Resolve DID:** The Lookup Service first resolves the asset's DID against the Verifiable Data Registry. From the returned DID Document, it retrieves two pieces of information: the Public Key for validation and the two service endpoints (one for the Dataspace Connector and one for the IoDT Verification Service).
2. **Fetch AAS Data:** The Lookup Service sends the initial request to the Dataspace Connector to retrieve the AAS. The Connector retrieves that data from the AAS Repository and returns it. At this point, the data is unverified and needs be to validated.
3. **Verify Asset:** After receiving the raw data, the Lookup Service initiates the integrity check by sending a Verification Challenge to the IoDT Verification Service.
   (a) The Verification Service performs the security check by delegating the challenge to the KMS, which signs it using the stored private key.
   (b) The Verification Service returns the resulting digital signature to the Lookup Service.
4. **Final Validation:** The Lookup Service validates the digital signature against the original Verification Challenge using the Public Key. This cryptographic proof confirms the identity of the data provider, assuring the user that the AAS data received in step 2 originates from the legitimate owner of the DT.

3.4.2 Semantic Interlinking Strategy

Implementing a framework capable of supporting the Circular Economy requires tracking a product throughout its lifecycle. This means a product's digital record must link back to the specific assets that created it, such as manufacturing machines or material batches, regardless of whether these assets reside within the same company or are distributed across other entities along the value chain. However, standard linking methods like Uniform Resource Locators (URLs) point to a specific server location. In industrial supply chains, servers and infrastructure may change frequently, causing these links to break.

To solve this, the IoDT proposes a semantic interlinking strategy based on the identity infrastructure defined in Section 3.4.1. Instead of pointing to a location, the DT points to the unique DID of the asset. The network can then resolve this DID to find the current location of the asset's data via the DID document, regardless of where it is hosted.

This architecture proposes a AAS Submodel named `IoDTInfo`. This submodel is designed to store all information about that asset that is related to the IoDT, including the network relationships for a DT. Eventually, as the IoDT architecture grows, it may store even more information.

The proposed structure of this metamodel is shown in Figure 3.8.

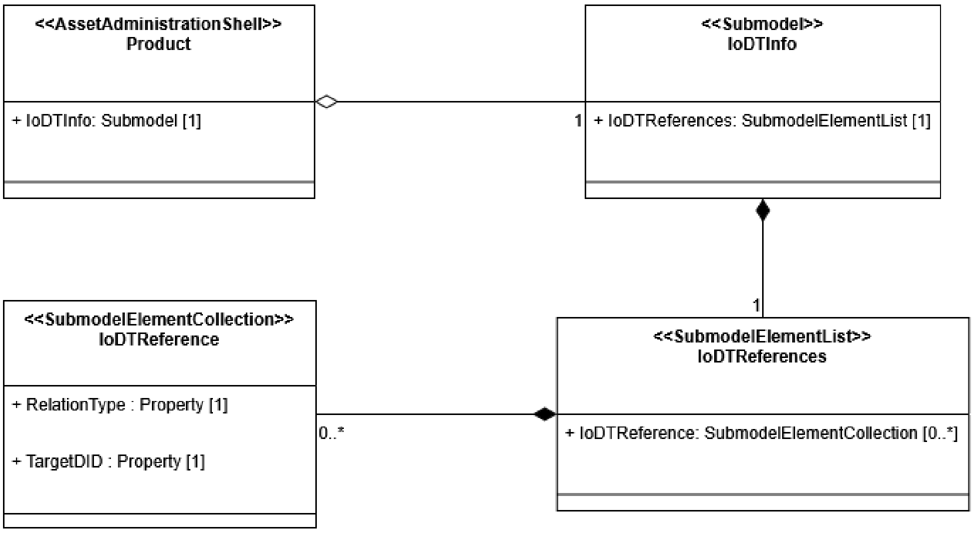


Figure 3.8: DID-based Semantic Interlinking Metamodel for the AAS

The metamodel utilizes the `SubmodelElementList` type from the AAS V3.0 metamodel[3] named `IoDTReferences`. Each entry in this list is defined as a `SubmodelElementCollection` which represents a single `IoDTReference` containing two mandatory fields:

- **RelationType:** Defines the semantic meaning of the link (e.g., `ManufacturedBy`, `ComponentOf`, `CertifiedBy`).
- **TargetDID:** The decentralized identifier of the referenced asset.

The following JSON snippet illustrates how this structure is realized within the AAS, highlighting the `globalAssetId` binding and the semantic reference to a manufacturing machine.

---

[3] https://aas-core-works.github.io/aas-core-meta/v3/

```
{
    ”assetAdministrationShells”: [
        {
            ”id”: ”https://example.com/ids/aas/product−123_shell”,
            ”modelType”: ”AssetAdministrationShell”,
            ”idShort”: ”Product_123_Shell”,
            ”assetInformation”: {
                ”assetKind”: ”Instance”,
                ”globalAssetId”: ”did:ebsi:zsSgDXeYPhZz3W...”,
            },
            ”submodels”: [
                {
                    ”type”: ”ModelReference”,
                    ”keys”: [
                        {
                            ”type”: ”Submodel”,
                            ”value”: ”https://example.com/ids/sm/iodt−info/12345”
                        }
                    ]
                }
            ]
        }
    ],
    ”submodels”: [
        {
            ”id”: ”https://example.com/ids/sm/iodt−info/12345”,
            ”modelType”: ”Submodel”,
            ”idShort”: ”IoDTInfo”,
            ”submodelElements”: [
                {
                    ”idShort”: ”IoDTReferences”,
                    ”modelType”: ”SubmodelElementList”,
                    ”typeValueListElement”: ”SubmodelElementCollection”,
                    ”value”: [
                        {
                            ”idShort”: ”Ref_001”,
                            ”modelType”: ”SubmodelElementCollection”,
                            ”value”: [
                                {
                                    ”idShort”: ”RelationType”,
                                    ”modelType”: ”Property”,
                                    ”valueType”: ”xs:string”,
                                    ”value”: ”ManufacturedBy”
                                },
                                {
                                    ”idShort”: ”TargetDID”,
                                    ”modelType”: ”Property”,
                                    ”valueType”: ”xs:string”,
                                    ”value”: ”did:ebsi:machine−001”
                                }
                            ]
                        }
                    ]
                }
            ]
        }
    ]
```

```
}
```

Listing 3.2: Example AAS with Semantic Interlinking

In conclusion, the architectural specifications detailed throughout this section provide a robust foundation for the IoDT's core network. By establishing a decentralized identity infrastructure and by defining a metamodel for asset interlinking, this proposal addresses the critical challenges of trust and traceability in a distributed and sustainable industrial ecosystem. While future work remains to fully implement advanced features such as VCs for granular claim verification, the current design incorporates the main requirements for a connected network of DTs supported by the DT Tier of the IoDT.

Chapter 4

Research Methodology

The preceding chapters have established the problem of industrial data interoperability and proposed a conceptual architecture, the IoDT, as a holistic solution. This chapter now defines the research methodology used to structure this research and to guide the evaluation of the artifacts produced. The goal is to describe how the research was conducted towards the development of IoDT artifacts and the evaluation of its utility.

4.1 Design Science Research

This work adopted the DSR paradigm. As defined by Hevner et al., the Information Systems discipline is characterized by two complementary paradigms. The behavioral science paradigm seeks to develop and verify theories that explain or predict human or organizational behavior, focusing on understanding reality. In contrast, the design science paradigm seeks to extend the boundaries of human and organizational capabilities by creating new and innovative artifacts. It is a problem-solving approach focused on changing reality by designing and building novel solutions in the form of constructs, models, methods, and instantiations. The primary goal of DSR is to generate knowledge through the process of building and evaluating these artifacts, thereby contributing to both the practical problem domain and the academic knowledge base [41].

4.1.1 The DSR Framework and Guidelines

Hevner et al. provide a conceptual framework for conducting DSR, which illustrates the interplay between the research activities and their environment. This framework (Figure 4.1), often referred to as the three-cycle view, highlights the core activities and the context in which they operate.

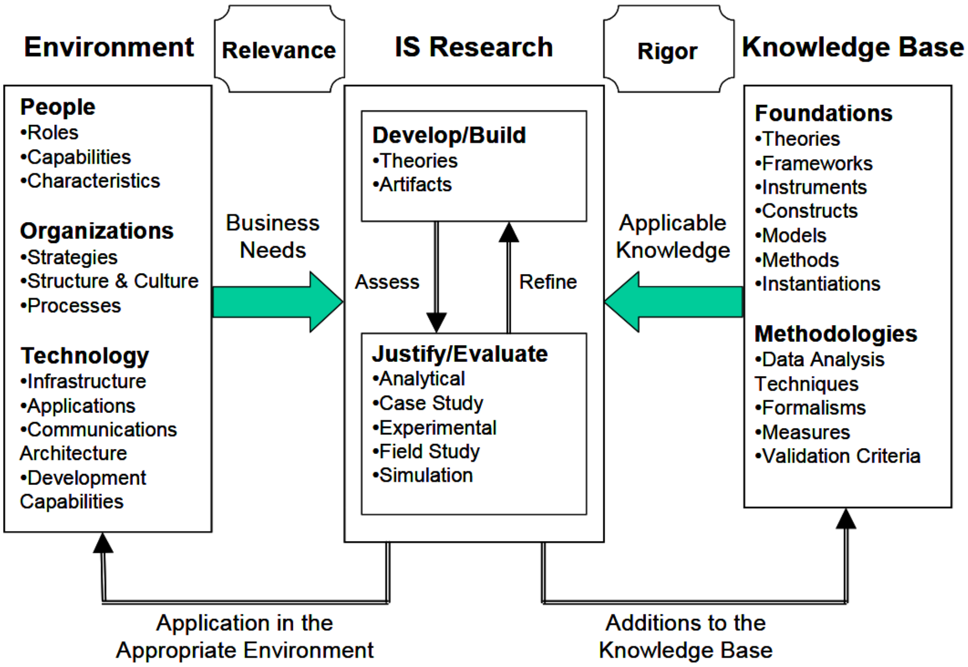


Figure 4.1: Information Systems Research Framework
Source: Hevner et al. [41]

The framework operates through three cycles that drive the research process. The Relevance Cycle bridges the Environment and the research activities; it takes business needs and problems as inputs and returns valid artifacts for field testing as outputs. The Rigor Cycle connects the Knowledge Base to the research, providing existing foundations, theories, and methodologies as inputs, while generating new knowledge and experiences as outputs. The central Design Cycle iterates between the construction and evaluation of the artifact, utilizing requirements from the Relevance Cycle and theories from the Rigor Cycle to refine the design.

The execution of these cycles follows an iterative and incremental approach. The artifacts are not developed in a single linear step but evolve through multiple iterations of the central Design Cycle. In each iteration, the understanding of the problem deepens, and the solution is incrementally improved based on feedback from the evaluation phase. This ensures that the final artifact effectively addresses the identified business needs while contributing to the scientific body of knowledge.

To guide the execution of high-quality DSR within this framework, Hevner et al. proposed seven fundamental guidelines (Table 4.1) that serve as the principles for conducting, evaluating, and sharing the research.

Guideline	Description
1. Design as an Artifact	DSR must produce a viable artifact (a construct, model, method, or instantiation).
2. Problem Relevance	The objective of DSR is to develop technology-based solutions to important and relevant business problems.
3. Design Evaluation	The utility, quality, and efficacy of the design artifact must be rigorously demonstrated via well-executed evaluation methods.
4. Research Contributions	Effective DSR must provide clear and verifiable contributions in the areas of the design artifact, its foundations, or its methodologies.
5. Research Rigor	DSR relies upon the application of rigorous methods in both the construction and evaluation of the artifact.
6. Design as a Search	The search for an effective artifact requires utilizing available means to reach desired ends while satisfying laws in the problem environment.
7. Communication	DSR must be presented effectively to both technology-oriented and management-oriented audiences.

Table 4.1: The Seven Guidelines of Design Science Research.
Adapted from Hevner et al. [41]

4.1.2 A Process-Oriented View of DSR

While the seven guidelines provide the foundational principles of DSR, their original presentation does not prescribe a specific chronological order. To make the methodology more actionable, scholars have interpreted these guidelines from a process perspective. Göbel and Cronholm analyzed the inputs and outputs of each guideline and proposed a holistic process model that arranges them in a logical sequence. This process-oriented view provides a clear roadmap for conducting a DSR project, from initial problem identification to the final communication of results. [42]

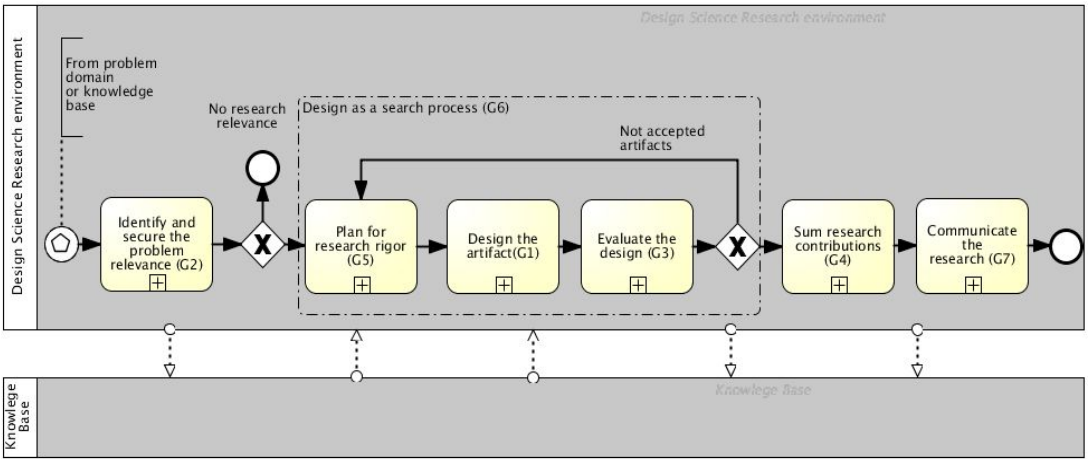


Figure 4.2: DSR Guidelines Illustration as a Process Model
Source: Göbel and Cronholm [42]

This model shows that a DSR project typically begins with Problem Relevance (G2) to ensure the work is meaningful. This is followed by Research Rigor (G5) to plan the project. The core of the work is the iterative Design as a Search Process (G6), which contains the build-and-evaluate cycles of Design as an Artifact (G1) and Design Evaluation (G3). Once an effective artifact is produced, the project moves to Research Contributions (G4) to consolidate the findings, and finally to Communication (G7) to disseminate the knowledge. The overall structure of this thesis aligns with this process-oriented interpretation of the DSR guidelines.

4.1.3 The Design Echelons Methodology

While a process view provides a valuable high-level structure for the research journey, this thesis adopts the echeloned DSR (eDSR) methodology proposed by Tuunanen et al. to manage the intricacies of each stage with a more granular organizing logic. This framework provides a modern and detailed approach for structuring DSR projects by decomposing them into a hierarchy of five self-contained ”design echelons.” Each echelon represents a logically coherent part of the research, produces a specific intermediate artifact, and has its own validation criteria, thus providing the specific framework for implementing the broader DSR process. [43]

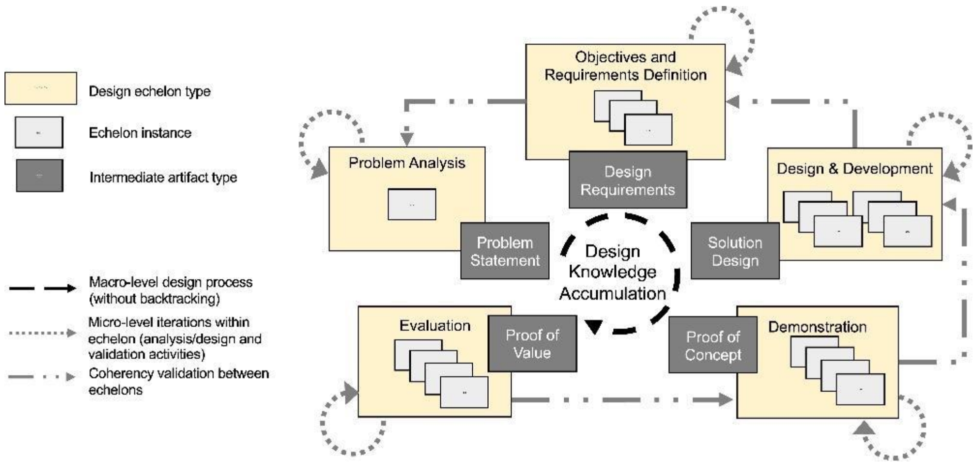


Figure 4.3: eDSR Metamodel
Source: Tuunanen et al [43]

The five design echelon types are:

1. **Problem Analysis Echelon:** Focuses on understanding and articulating the research problem.
2. **Objectives and Requirements Definition Echelon:** Defines the goals for a solution.
3. **Design and Development Echelon:** Creates a projectable solution design for the artifact.
4. **Demonstration Echelon:** Instantiates the artifact to prove its functional feasibility (Proof of Concept.

5. **Evaluation Echelon:** Assesses the contextualized artifact in use to prove its value (Proof of Value).

The structure of this thesis has been explicitly mapped to this methodology. Chapters 1 and 2 fulfilled the Problem Analysis Echelon. Chapter 3 fulfilled the Design and Development Echelon. Chapter 5 is dedicated to the Demonstration and Evaluation Echelons, where the feasibility and utility of the IoDT artifacts will be experimentally validated.

4.2 Design Artifacts and Instantiation of the DSR Process

This research followed the eDSR methodology to produce two primary design artifacts that serve as the main contributions of this thesis:

- **Artifact 1 (Conceptual):** The IoDT Architecture. This is the primary contribution of the thesis, a conceptual model and holistic framework that integrates existing standards into a cohesive, three-tiered architecture.
- **Artifact 2 (Implementation):** A Proof-of-Concept of the Data Tier. This is a tangible instantiation that serves to validate the foundational layer of the conceptual architecture.

The development of these artifacts was executed through the five design echelons of the eDSR framework, grounded in the practical context of the PRODUTECH R3 project. Specifically, the validation phase (Echelons 4 and 5) follows an "experiments" approach, as suggested by Mettler et al. [44], where inputs are manipulated and outputs are observed to demonstrate the utility, feasibility, and impact of the artifacts.

1. **Problem Analysis Echelon:** The process began with a deep analysis of the problem domain. This involved iterations between a comprehensive literature review on the data challenges of the Circular Economy and I4.0, and conversations with industry experts. These discussions typically occurred during PRODUTECH R3 project meetings.
2. **Objectives and Requirements Definition Echelon:** Based on the problem analysis, the specific objectives for a solution were defined. The primary requirement was to establish a holistic architecture capable of supporting the circular economy and the DPP, bridging the gap between OT/IT integration and data interoperability.
3. **Design and Development Echelon:** This echelon focused on the creation of the solution artifact: the IoDT Reference Architecture. The design process involved iteratively modeling the three tiers of the architecture to ensure compliance with the RAMI 4.0 framework.
4. **Demonstration Echelon:** To demonstrate the technical feasibility of the proposed design, a Proof of Concept was implemented (Experiment 1). This involved a practical implementation of the architecture's "Data Tier."
5. **Evaluation Echelon:** The final echelon assessed the utility of the complete architecture through a Proof of Value (Experiment 2). This evaluation was conducted using a scenario-based approach, applying the IoDT architecture to realistic industrial use cases derived from the PRODUTECH R3 context.

Chapter 5

Experimental Validation of the IoDT Architecture

The preceding chapters have established the problem of industrial data interoperability, presented the IoDT as a conceptual solution, and defined the DSR methodology for its validation. This chapter now executes that methodology, presenting the results of two experiments designed to evaluate the design artifacts produced in this thesis. Following the Design Echelons framework, this chapter addresses the final two echelons. The first experiment serves as the Demonstration Echelon, providing a Proof of Concept to demonstrate the technical feasibility of the implemented artifact. The second experiment serves as the Evaluation Echelon, providing a Proof of Value to argue for the utility of the conceptual architecture.

5.1 Experimental Demonstration of the Data Tier Interoperability (Proof of Concept)

This experiment is designed to fulfill the Demonstration Echelon of the eDSR methodology. Its primary objective is to demonstrate the technical feasibility of the foundational layer of the IoDT architecture by tackling the first two challenges identified in Chapter 2: the integration and standardization of data from disparate OT and IT sources. The artifact under evaluation for this experiment is Artifact 2, the Proof of Concept implementation of the Data Tier.

The experiment is divided into two parts, each addressing a different data origin: the first focuses on OT data from machinery (section 5.1.1), and the second on IT data from an ERP (section 5.1.2). It explores both manual and automated AAS data structuring. A more detailed technical breakdown of this experiment is available in Appendix A.

5.1.1 OT Data Integration

This experiment demonstrates the automated generation of an AAS from simulated machine sensor data. The technical setup introduces several open-source technologies:

- **Node-RED:** A tool used to simulate real-time data streams from two temperature sensors housed within a single machine.

- **Eclipse Mosquitto:** A Message Queuing Telemetry Transport (MQTT) Broker, the messaging protocol that served for transmitting the sensor data.
- **Eclipse Ditto:** A Digital Twin management platform used to manage the state of a large number of digital ”Things” and provide a rich API for interacting with them. ”Things”, according to Ditto’s conceptual view are generic entities and are mostly used as a “handle” for multiple features belonging to this ”Thing”. In this experiment, a ”Thing” was created for each sensor, organized under a common namespace representing the machine.
- **Eclipse BaSyx:** The target AAS repository.

To clearly demonstrate the data integration pipeline, the process is detailed in Table 5.1 as a Use Case Scenario. This workflow effectively functions as an Extract-Transform-Load (ETL) process, moving data from the operational side to the AAS repository.

Title	**Automated AAS Generation from OT Data**
Pre-conditions	1. A digital ”Thing” representing the sensor exists in Eclipse Ditto. 2. The MQTT Broker is active. 3. Node-RED is configured to publish data. 4. The Connections from Eclipse Ditto to both the AAS Repository and MQTT Broker are configured.
Steps	1. **Extraction:** Node-RED publishes simulated temperature readings to the MQTT broker. 2. **Ingestion:** Eclipse Ditto’s Inbound Connection subscribes to the topic. 3. **Transformation 1:** The Inbound Payload Mapper converts raw data into Ditto’s ”Thing” model. 4. **Internal Update:** The ”Thing” state is updated. 5. **Trigger:** Outbound Connection detects state change. 6. **Transformation 2:** Outbound Mapper converts ”Thing” data to AAS format. 7. **Load:** Ditto pushes the payload to the BaSyx API.
Post-conditions	The Asset Administration Shell in BaSyx is updated with real-time data.

Table 5.1: Use Case Scenario: Automated AAS Generation from OT Data

This process began with the prerequisite step of creating a ”Thing” in Eclipse Ditto for each simulated sensor, organizing them under a common namespace to represent the machine. Once this digital representation was established, the real-time data flow proceeded as follows: Node-REDConcl published simulated temperature readings to the MQTT broker. The integration was orchestrated using Eclipse Ditto’s Connectivity Service, which managed both inbound and outbound data flows. An inbound MQTT connection was configured to subscribe to the broker’s topic, and its payload mapper transformed the incoming sensor data into Ditto's "Thing" data model, updating the state of the digital twin. Following this, and inspired by the integration pattern described by Kristan et al. [45], an outbound HTTP connection listened for these state changes. Its payload mapper transformed the "Thing" data into the standardized AAS format and pushed the result to the BaSyx repository's API. This, in turn, created and updated the corresponding AASs for each sensor. Figure 5.1 further demonstrates the interactions between the different components in this workflow through a sequence diagram.

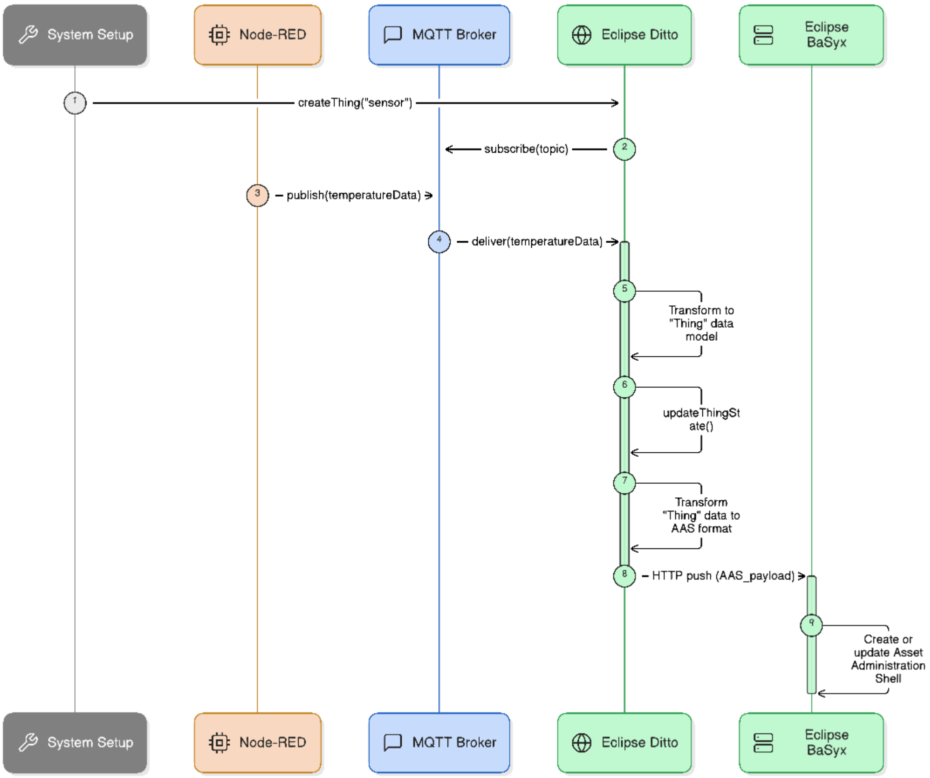


Figure 5.1: Sequence Diagram for OT Data Integration with Eclipse Ditto

The successful execution of this experiment resulted in the automated generation and continuous updating of AASs in the BaSyx server based on real-time OT data streams. This demonstrates a feasible and scalable pathway for standardizing machine-level data.

5.1.2 IT Data Integration

This experiment demonstrates the automated generation of an AAS from an IT enterprise system using an Artificial Intelligence (AI) Agent, orchestrated by a Large Language Model (LLM), to perform complex tasks. The technical setup consisted of:

- **Odoo:** An ERP containing product data.
- **Model Context Protocol (MCP) Servers:** Standardized interfaces that exposed the functionalities of Odoo and BaSyx as tools to the AI Agent. While a pre-existing open-source MCP server was available for Odoo[1], a simple, proof of

---

[1] `https://github.com/tuanle96/mcp-odoo`

concept MCP server for Eclipse BaSyx was developed as part of this research to enable the integration.

- **AI Agent:** An orchestrator powered by an LLM (Claude AI) capable of understanding natural language prompts and using the MCP servers as tools to interact with other systems.
- **Eclipse BaSyx:** The target repository for the generated AAS.

The experiment, depicted in Figure 5.2, was executed by providing the AI Agent with natural language prompts, asking it to fetch a product from Odoo and then generate an AAS to store it in BaSyx. The agent's underlying LLM interpreted the prompts and executed them by calling the appropriate tools: it first queried the Odoo MCP server to retrieve the product data, then constructed a valid AAS payload, and finally used the BaSyx MCP server to create the new AAS in the repository. The successful result was a product AAS generated from a simple, high-level command.

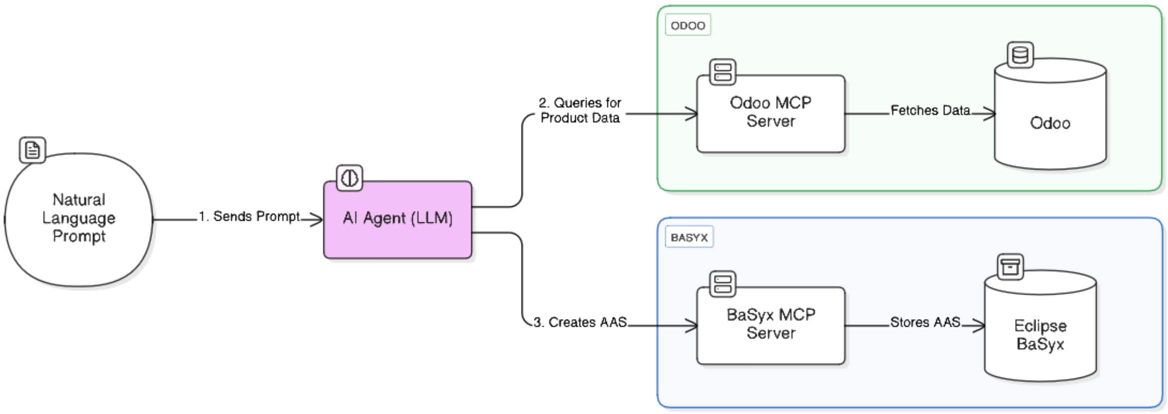


Figure 5.2: AI-assisted IT Data Integration Architecture

In summary, the goal was to streamline the creation of AAS data structures using AI, wherein the AI agent is queried in natural language and responds in "AAS mode".

Furthermore, this integration pattern demonstrates dual utility. Beyond the automated creation of AASs, the same architecture of an AI Agent connected to the BaSyx repository via an MCP server can be leveraged for natural language querying. This allows stakeholders to easily interrogate the industrial data ecosystem without needing to understand the underlying technical schemas. For instance, a manager could ask sustainability-focused questions such as, "Which products in the repository contain over 40% recycled materials?", "What is the carbon footprint of the batch with ID 123?", or "List all components that have a valid FSC certification[2]." This capability significantly lowers the barrier to accessing and utilizing critical sustainability data, making the information more actionable for decision-making and compliance.

---

[2] The Forest Stewardship Council (FSC) certification is an international standard that verifies wood and paper products are sourced from responsibly managed forests that provide environmental, social, and economic benefits

5.1.3 Conclusion of Experiment 1

Together, these two sub-experiments provide a strong Proof of Concept for the Data Tier of the IoDT architecture. They successfully demonstrate that the proposed approach is technically feasible for addressing the first and second hurdles of industrial interoperability: data integration and standardization. The experiments show that data can be successfully consolidated from both the OT and IT layers and transformed into the mandatory, standardized format of the AAS. This validates the design of the Data Tier as a practical solution, demonstrating that the integration of these open-source technologies can successfully automate tasks that are typically manual, laborious, and resource-intensive.

However, a potential disadvantage of using the MCP in this context is the intrinsic lack of control over the data flow, which relies heavily on the LLM's interpretation. This limitation suggests that while the AI-driven approach offers flexibility, it may be replaced by more deterministic approaches based on ontologies.

5.2 Evaluating the Utility of the IoDT Architecture (Proof of Value)

This experiment is designed to fulfill the Evaluation Echelon of the eDSR methodology. It is a scenario-based "thought experiment" with the objective of evaluating the utility of the complete conceptual architecture for the IoDT (Artifact 1) and arguing for its value as a comprehensive solution.

This research work was conducted within the scope of the PRODUTECH R3 - "Agenda Mobilizadora da Fileira das Tecnologias de Produção para a Reindustrialização" project, an initiative aimed at fostering reindustrialization through advanced production technologies. This research contributes specifically to the project's Work Package 8 (WP8), "Digital&Autonomous Factory" which focuses on developing solutions for a more connected, intelligent, and sustainable industrial ecosystem capable of accommodating high product mix and demand fluctuations without compromising operational and environmental efficiency [46].

To ground the evaluation in a realistic context, the following scenarios are directly inspired by the business realities of the PRODUTECH R3 project's partner companies, MUVU Technologies and Plasoeste. This approach ensures that the validation is not merely a theoretical exercise but is directly relevant to the opportunities and challenges faced by the industry today, thereby fulfilling the "Relevance Cycle" of DSR. Before the description of the experiment, a brief presentation of the involved partners is given:

- **MUVU Technologies:** A SME that develops high-tech solutions for smart industries, including their flagship MES product, RAILES. MUVU operates in a world of high product customization and dynamic production lines. For a company in their position, the ability to provide transparent, real-time information about material provenance, sustainability certifications, and product lifecycle data represents a significant opportunity to gain a competitive advantage and prepare for future regulatory compliance [47].
- **Plasoeste:** A company specializing in the flexible plastics packaging industry since 1986. Plasoeste manufactures a wide range of products, including trash bags, food-

grade films, and industrial sleeves, for a diverse Business-to-Business (B2B) client base across sectors such as agrifood, pharmaceutical, hospitality, and waste management. With a strong, publicly stated commitment to sustainability, evidenced by multiple certifications, Plasoeste's industrial context presents a clear opportunity to enhance their operations by verifiably tracking and communicating sustainability metrics to their industrial clients and for future regulatory compliance [48].

The following scenarios are hypothetical but are directly inspired by the business contexts of these companies to demonstrate the architecture's relevance to real-world industrial challenges.

5.2.1 Dynamic and Verifiable Subcontracting via MES

Scenario: A client of MUVU's MES product, RAILES, manufactures complex, customizable products. An urgent order requires a specific component to be manufactured using a "low-carbon" certified process. The client's own factory is at full capacity and cannot fulfill this step internally. They need a way to dynamically subcontract this production step to a trusted external partner while ensuring that the sustainability claim is verifiable and that all production data is seamlessly integrated back into the main production order within their MES.

Table 5.2 provides an analysis of how each tier of the IoDT architecture contributes to the proposed solution.

IoDT Tier	Analysis
APP Tier	A SMaaS marketplace, acting as an application within the IoDT, is queried through the RAILES MES to find partners. The marketplace searches the data space for partners possessing the required technical capability and the specific, verifiable "low-carbon process" credential.
DT Tier	The MES discovers a potential partner whose factory's AAS holds a VC for this process, issued by a trusted third-party auditor. The MES then cryptographically verifies this VC by resolving the auditor's DID and validating the signature.
Data Tier	Once the partner is verified and selected, the MES shares a part of the main production order with the partner via the data space. The partner's machines, integrated via their own Data Tier, execute the production, sending real-time status updates back to the RAILES MES as AAS updates.

Table 5.2: IoDT Tier Analysis for the MUVU-Inspired Scenario

The production step is completed by a trusted external partner. The sustainability requirement is met with cryptographic certainty, and all relevant data (e.g., material usage, energy consumption, quality control) is seamlessly integrated back into the client's master record for the product's DPP. This demonstrates how the IoDT architecture extends a traditional MES from a single-factory management tool into a sustainable multi-enterprise production orchestrator.

5.2.2 Verifiable Closed-Loop Recycling Partnership

Scenario: A company like Plasoeste has a partnership with a large hotel chain, supplying them with trash bags made from recycled plastic. To enhance their joint sustainability goals, they establish a closed-loop recycling program. The hotel agrees to segregate its plastic waste, which Plasoeste then collects to use as raw material for new products. The challenge is to create an auditable data trail that proves this circularity, ensuring the hotel can confidently report its participation in a circular economy, and Plasoeste can prove the usage of recycled materials.

Table 5.3 provides an analysis of how each tier of the IoDT architecture enables this dynamic, circular process.

IoDT Tier	Analysis
Data Tier	**At the Hotel:** When a bale of used plastic is ready for collection, the hotel's waste management system (integrated via its Data Tier) automatically generates an AAS for this batch of waste, including its weight, collection date, and origin. **At Plasoeste:** The Data Tier integrates with their ERP and MES to track when these specific batches of waste coming from the Hotel are consumed in a production run, creating an AAS for the new batch of bags.
DT Tier	Each AAS (for both the waste and the new product) is assigned a unique DID. When the new batch AAS is created, it utilizes the semantic interlinking strategy defined in the architecture. The `IoDTReferences` collection within the `IoDTInfo` Submodel is updated with multiple `IoDTReference` entries that link it to the DIDs of the waste batches used in its production." An auditor, who has verified this data pipeline, issues a VC attached to the new batch's AAS, attesting to the verifiable circularity and recycled content percentage.
APP Tier	The hotel shares the waste batches AASs with Plasoeste via a data space. In return, Plasoeste shares the AAS for the new batches of bags. The hotel can then access this data to verify the circularity claims and generate accurate data for its sustainability reports.

Table 5.3: IoDT Tier Analysis for the Plasoeste-Inspired Scenario

This transforms a simple business agreement into a transparent, data-driven circular ecosystem. The hotel receives cryptographic proof that their waste is being recycled into new products, providing them with auditable data for their sustainability reporting. Plasoeste gains a verifiable record of their raw material provenance, strengthening the integrity of their sustainability claims. This scenario demonstrates the IoDT's capacity to facilitate circular economy models that depend on multi-party trust and data lineage.

5.2.3 Conclusions for Experiment 2

These scenarios provide the Proof of Value for the IoDT architecture. While the necessity for industrial data sharing, soon to be enforced by regulations like the DPP, highlights the value of the IoDT, its utility extends beyond simple compliance. The scenarios demonstrate that, unlike fragmented solutions, the IoDT provides a complete, end-to-end process that enables advanced applications. By integrating all the necessary components to solve these real industrial problems, the architecture proves its value as a superior solution for building a reliable, data-driven, and sustainable industrial ecosystem.

Chapter 6

Conclusions and Future Work

The transition to a data-driven Circular Economy, accelerated by regulatory mandates like the DPP, has created an urgent need for a cohesive solution to the long-standing problem of industrial data fragmentation. This thesis addressed this challenge by proposing the IoDT, a holistic conceptual architecture designed to serve as a practical guidebook for implementing interoperable, sovereign, and sustainable industrial data ecosystems.

The primary contribution of this work is the IoDT architecture itself, a RAMI 4.0-compliant blueprint that integrates standards for data modeling (AAS), sovereign data sharing (IDS), and a decentralized trust layer (DIDs and VCs) with semantic asset interlinking. By tracing the journey of industrial data, this thesis identified a 3D Data Challenge, with multiple obstacles that needed to be overcome. The need for a holistic architecture that directly addressed each of those challenges was demonstrated, and that is the goal of creating the concept of the IoDT.

Following a DSR approach, the validation presented in Chapter 5 confirmed the viability of this approach. The Proof of Concept successfully demonstrated the technical feasibility of the Data Tier, proving that data from heterogeneous OT and IT sources can be automatically integrated and standardized into the AAS format using modern, open-source technologies. Subsequently, the Proof of Value, through realistic industrial scenarios, argued for the utility of the full architecture. It demonstrated that the IoDT not only provides a foundation for regulatory compliance (DPP) but also enables advanced applications such as dynamic subcontracting for sustainable manufacturing and verifiable circular economy partnerships.

In conclusion, this thesis has successfully delivered on its primary goal: to present and justify an architectural framework that can guide industries in building the secure, interoperable, and trustworthy data ecosystems required to participate in the next generation of sustainable manufacturing. While the work establishes this robust conceptual foundation and demonstrates the feasibility of the IoDT's foundational layer, the IoDT remains, for the most part, a conceptual architecture. The logical path forward is to continue its development and validate it through real-world implementation. Future work should focus on expanding the existing Data Tier proof of concept and implementing the remaining DT and APP Tiers to create a full-stack prototype. The ultimate goal is to deploy this prototype in a real-world case study, ideally in partnership with an industrial company, to gather evidence of the architecture's actual impact in the industrial world.

List of Author's Publications

The following work was produced and accepted for publication during the course of this research:

- D. Neto, C. Sousa, C. Pereira, R. Ferreira, and P. Pinto, "Architecting Circularity through an Internet of Digital Twins Framework," in *Proceedings of the 20th Iberian Conference on Information Systems and Technologies (CISTI 2025)*, 2026. (To be published)

Bibliography

[1] M. Hermann, T. Pentek, and B. Otto, “Design principles for industrie 4.0 scenarios,” in *2016 49th Hawaii International Conference on System Sciences (HICSS)*, pp. 3928–3937, 2016.

[2] E. Commission, D.-G. for Research, Innovation, M. Breque, L. De Nul, and A. Petridis, *Industry 5.0 – Towards a sustainable, human-centric and resilient European industry.* Publications Office of the European Union, 2021.

[3] M. Geissdoerfer, P. Savaget, N. M. Bocken, and E. J. Hultink, “The circular economy – a new sustainability paradigm?,” *Journal of Cleaner Production*, vol. 143, pp. 757–768, 2017.

[4] European Commission, “Proposal for a regulation on ecodesign for sustainable products.” `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52022PC0142`, March 2022.

[5] J. M. Müller, O. Buliga, and K.-I. Voigt, “Fortune favors the prepared: How smes approach business model innovations in industry 4.0,” *Technological Forecasting and Social Change*, vol. 132, pp. 2–17, 2018.

[6] K. Nagorny, P. Monteiro, J. Barata, and A. Colombo, “Big data analysis in smart manufacturing: A review,” *International Journal of Communications, Network and System Sciences*, vol. 10, pp. 31–58, 03 2017.

[7] E. Kristoffersen, F. Blomsma, P. Mikalef, and J. Li, “The smart circular economy: A digital-enabled circular strategies framework for manufacturing companies,” *Journal of Business Research*, vol. 120, pp. 241–261, 2020.

[8] M. Farooque, A. Zhang, M. Thürer, T. Qu, and D. Huisingh, “Circular supply chain management: A definition and structured literature review,” *Journal of Cleaner Production*, vol. 228, pp. 882–900, 2019.

[9] F. Takacs, D. Brunner, and K. Frankenberger, “Barriers to a circular economy in small- and medium-sized enterprises and their integration in a sustainable strategic management framework,” *Journal of Cleaner Production*, vol. 362, p. 132227, 2022.

[10] J. Walden, A. Steinbrecher, and M. Marinkovic, “Digital product passports as enabler of the circular economy,” *Chemie Ingenieur Technik*, vol. 93, no. 11, pp. 1717–1727, 2021.

[11] W. Kritzinger, M. Karner, G. Traar, J. Henjes, and W. Sihn, “Digital twin in manufacturing: A categorical literature review and classification,” *IFAC-PapersOnLine*,

vol. 51, no. 11, pp. 1016–1022, 2018. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.

[12] M. Wollschlaeger, T. Sauter, and J. Jasperneite, “The future of industrial communication: Automation networks in the era of the internet of things and industry 4.0,” *IEEE Industrial Electronics Magazine*, vol. 11, no. 1, pp. 17–27, 2017.

[13] J. Schlechtendahl, M. Keinert, F. Kretschmer, A. Lechler, and A. Verl, “Making existing production systems industry 4.0-ready: Holistic approach to the integration of existing production systems in industry 4.0 environments,” *Production Engineering*, vol. 9, p. 143–148, Oct. 2014.

[14] W. Mahnke, S.-H. Leitner, and M. Damm, *OPC Unified Architecture*. Springer Berlin Heidelberg, 2009.

[15] Y. Lu and X. Xu, “Cloud-based manufacturing equipment and big data analytics to enable on-demand manufacturing services,” *Robotics and Computer-Integrated Manufacturing*, vol. 57, pp. 92–102, 2019.

[16] M. S. Sodhi and C. S. Tang, “Research opportunities in supply chain transparency,” *Production and Operations Management*, vol. 28, no. 12, pp. 2946–2959, 2019.

[17] A. Sharma, E. Kosasih, J. Zhang, A. Brintrup, and A. Calinescu, “Digital twins: State of the art theory and practice, challenges, and open research questions,” *Journal of Industrial Information Integration*, vol. 30, p. 100383, 2022.

[18] M. Jäger-Roschko and M. Petersen, “Advancing the circular economy through information sharing: A systematic literature review,” *Journal of Cleaner Production*, vol. 369, p. 133210, 2022.

[19] M. Jurmu, I. Niskanen, A. Kinnula, J. Kääriäinen, M. Ylikerälä, P. Räsänen, and T. Tuikka, “Exploring the role of federated data spaces in implementing twin transition within manufacturing ecosystems,” *Sensors*, vol. 23, no. 9, 2023.

[20] L. Nagel and D. Lycklama, “Design principles for data spaces - position paper,” July 2021.

[21] S. Ritzén and G. Ölundh Sandström, “Barriers to the circular economy – integration of perspectives and domains,” *Procedia CIRP*, vol. 64, pp. 7–12, 2017. 9th CIRP IPSS Conference: Circular Perspectives on PSS.

[22] Industrial Data Spaces Association (IDSA), “Ids reference architecture model 4.” `https://docs.internationaldataspaces.org/ids-knowledgebase/ids-ram-4`.

[23] F. Schöppenthau, F. Patzer, B. Schnebel, K. Watson, N. Baryschnikov, B. Obst, Y. Chauhan, D. Kaever, T. Usländer, and P. Kulkarni, “Building a digital manufacturing as a service ecosystem for catena-x,” *Sensors*, vol. 23, no. 17, 2023.

[24] B. D. Allen, “Digital twins and living models at nasa,” in *Digital Twin Summit*, 2021.

[25] E. Negri, L. Fumagalli, and M. Macchi, “A review of the roles of digital twin in cps-based production systems,” *Procedia Manufacturing*, vol. 11, pp. 939–948, 2017. 27th International Conference on Flexible Automation and Intelligent Manufacturing, FAIM2017, 27-30 June 2017, Modena, Italy.

[26] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Nee, “Enabling technologies and tools for digital twin,” *Journal of Manufacturing Systems*, vol. 58, pp. 3–21, 2021. Digital Twin towards Smart Manufacturing and Industry 4.0.

[27] M. Abbas, Y. Liu, and B. Cai, “Digital twin-based prognostics and health management for subsea systems: Concepts, classification, opportunities and challenges,” in *Proceedings of the 31st European Safety and Reliability Conference (ESREL 2021)*, ESREL, p. 1697–1703, Research Publishing Services, 2021.

[28] F. Tao, M. Zhang, Y. Liu, and A. Nee, “Digital twin driven prognostics and health management for complex equipment,” *CIRP Annals*, vol. 67, no. 1, pp. 169–172, 2018.

[29] J. Monteiro, J. Barata, and S. Gentilini, “A digital twin-based digital product passport,” *Procedia Computer Science*, vol. 246, pp. 4123–4132, 2024. 28th International Conference on Knowledge Based and Intelligent information and Engineering Systems (KES 2024).

[30] A. Ventura, C. Sousa, C. Pereira, N. Duarte, M. Martins, and B. Silva, “Modelling circular-driven digital twins,” *Procedia Computer Science*, vol. 256, pp. 617–623, 2025. CENTERIS - International Conference on ENTERprise Information Systems / ProjMAN - International Conference on Project MANagement / HCist - International Conference on Health and Social Care Information Systems and Technologies.

[31] T. A. Abdel-Aty, E. Negri, and S. Galparoli, “Asset administration shell in manufacturing: Applications and relationship with digital twin,” *IFAC-PapersOnLine*, vol. 55, no. 10, pp. 2533–2538, 2022. 10th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2022.

[32] Industrial Digital Twin Association (IDTA), “Asset administration shell specifications.” `https://industrialdigitaltwin.io/aas-specifications/index/home/index.html`, 2025. Accessed 2025.

[33] L. Kunz, M. Reif, T. Petzsche, A. Schmallenbach, C. Plociennik, and M. Ruskowski, “Realizing closed-loop supply chain networks based on dataspaces and manufacturing marketplaces,” *IFAC-PapersOnLine*, vol. 58, no. 19, pp. 235–240, 2024. 18th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2024.

[34] E. Bakopoulos, K. Sipsas, N. Nikolakis, and K. Alexopoulos, “A digital twin and data spaces framework towards resilient manufacturing value chains,” *IFAC-PapersOnLine*, vol. 58, no. 19, pp. 163–168, 2024. 18th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2024.

[35] Y. Xia, N. Jazdi, and M. Weyrich, “Automated generation of asset administration shell: a transfer learning approach with neural language model and semantic fingerprints,” in *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*, pp. 1–4, 2022.

[36] A. Ricci, A. Croatti, S. Mariani, S. Montagna, and M. Picone, “Web of digital twins,” *ACM Trans. Internet Technol.*, vol. 22, Nov. 2022.

[37] Plattform Industrie 4.0, “RAMI4.0 – a reference framework for digitalisation.” `https://www.plattform-i40.de/IP/Redaktion/EN/Downloads/Publikation/rami40-an-introduction.html`, August 2018.

[38] World Wide Web Consortium, “Decentralized Identifiers (DID) v1.0.” `https://www.w3.org/TR/did-1.0/`, 2022.

[39] World Wide Web Consortium, “Verifiable Credentials Data Model v2.0.” `https://www.w3.org/TR/vc-data-model-2.0/`, 2025.

[40] M. Neubauer, L. Steinle, C. Reiff, S. Ajdinović, L. Klingel, A. Lechler, and A. Verl, “Architecture for manufacturing-x: Bringing asset administration shell, eclipse dataspace connector and opc ua together,” *Manufacturing Letters*, vol. 37, pp. 1–6, 2023.

[41] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” *MIS Quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[42] H. Göbel and S. Cronholm, “Design science research in action : experiences from a process perspective,” 2012.

[43] T. Tuunanen, R. Winter, and J. v. Brocke, “Dealing with complexity in design science research: A methodology using design echelons1,” *Management Information Systems Quarterly*, vol. 48, pp. 427–458, 06 2024.

[44] T. Mettler, M. Eurich, and R. Winter, “On the use of experiments in design science research: A proposition of an evaluation framework,” *Communications of the Association for Information Systems*, vol. 34, pp. 223–240, 01 2014.

[45] J. Kristan, M. Jäntgen, S. E. Jeroschewski, and M. Grzanna, “Access ditto things from an asset administration shell.” `https://eclipse.dev/ditto/2024-02-27-integrating-ditto-aas-basyx.html`, feb 2024.

[46] PRODUTECH R3, “WP8 – Digital&Autonomous Factory.” `https://r3.produtech.org/en/work-package/wp8-digitalautonomous-factory-fabrica-digital-e-sustentavel-7`.

[47] “MUVU Technologies.” `https://muvu.tech/`.

[48] “Plasoeste - Embalagens Plásticas Flexíveis.” `https://plasoeste.com/`.

Appendix A

Technical Details for Experiment 1

This appendix explores the key technologies involved and the implementation details of the two parts of Experiment 1, conducted to demonstrate the technical feasibility of the IoDT's Data Tier.

A.1 Technologies Involved

A.1.1 Eclipse BaSyx

Eclipse BaSyx[1] is an open-source middleware platform for I4.0, designed to be one of the first and most comprehensive implementations of the AAS as a standardized DT. Its mission is to provide a free, re-usable, and off-the-shelf set of software components and Software Development Kits (SDKs) that enable companies of all sizes to develop and implement I4.0 solutions.

It provides all the necessary infrastructure components for creating and managing an AAS ecosystem. The key components relevant to this experiment include:

- **AAS Server:** A server component responsible for hosting the AASs and their Submodels. It exposes an API for managing AAS information, serving as the central repository for the DTs.
- **AAS Registry:** A discovery service that allows applications and users to find available AASs. It maintains a list of all registered AASs and the endpoints for their respective AAS Servers.
- **DataBridge Component:** BaSyx also offers a DataBridge component for connecting directly to industrial data sources and protocols like OPC UA or MQTT.

Eclipse BaSyx served as the target repository for this experiment, allowing the standardized data from both OT and IT systems to be stored, managed, and accessed in a uniform manner.

---

[1] `https://www.eclipse.org/basyx/`

A.1.2 Eclipse Ditto

Eclipse Ditto[2] is an open-source framework for creating and managing digital twins in the context of the Internet of Things. It is designed to be domain-agnostic, meaning it can be applied to various sectors, including industrial and residential domains. Ditto provides a unified interface and data model to abstract the complexity of interacting with a heterogeneous landscape of physical devices.

At its core, Ditto is built on a microservice architecture, with several key services working together, as illustrated in Figure A.1.

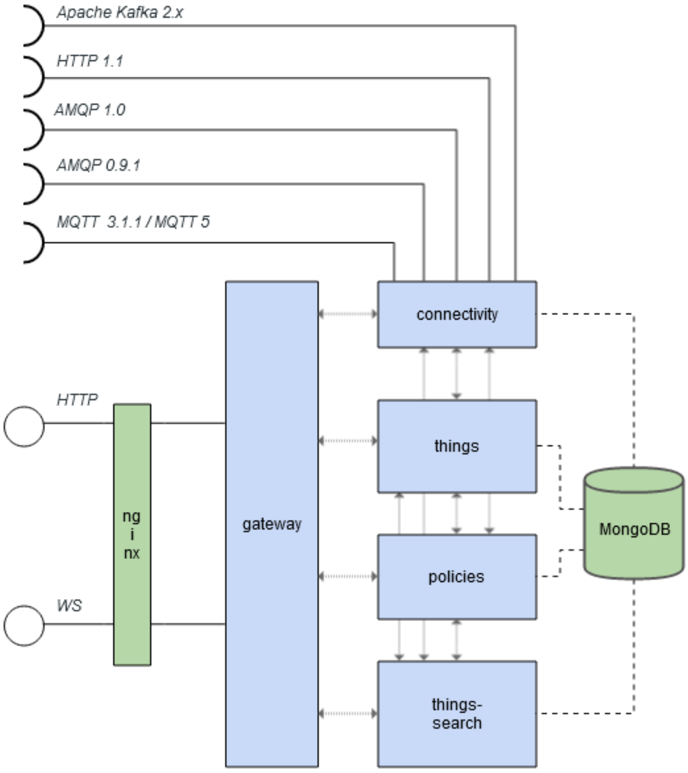


Figure A.1: Eclipse Ditto Architecture Overview
Source: Eclipse Ditto

The most relevant service for this experiment is the Connectivity Service, which is responsible for all external communication. It manages the data exchange between physical devices and their digital representations. The core concepts of Ditto's domain model are:

- **Thing:** The central entity in Ditto, representing the digital twin of a physical or virtual asset (e.g., a sensor, a machine, or even a process). Each Thing is uniquely identified by a `thingId`, which consists of a `namespace` and a `name`.
  - A `thingId` consists of a Namespace and a Name (e.g., `my.namespace:my-thing-name`).
- **Policy:** A separate entity that defines fine-grained access control. Each Thing is linked to a Policy via its `policyId`. A Policy contains a set of entries, where each entry maps Subjects (who has access) to Resources (what part of a Thing is being accessed) with specific grant or revoke permissions for `READ` and `WRITE` operations.

---

[2] `https://www.eclipse.dev/ditto/`

- **Attributes:** A JSON object for storing static metadata about a Thing, such as a serial number or manufacturer, which does not change frequently.
- **Features:** Logical groups of related state data. For example, a sensor might have a ”temperature” feature that contains all temperature-related properties.
- **Properties:** The actual data values within a Feature that represent the state of the asset, such as the current temperature reading.

The Connectivity Service orchestrates data flow through Connections. A Connection defines a communication channel to an external system and is configured with sources for inbound data and targets for outbound data. The transformation of data in both directions is handled by Payload Mappers. These are typically JavaScript scripts that translate between different data formats. An inbound mapper translates data from a device's native format into Ditto's internal Thing model, while an outbound mapper translates data from the Thing model into the format required by an external system, such as the AAS format for the BaSyx API.

For this experiment, Ditto was intentionally chosen as the intermediary for OT connectivity over the native BaSyx DataBridge. While the BaSyx DataBridge is a capable component for creating data pipelines, it functions primarily as an ETL (Extract, Transform, Load) tool focused on data transport from a source to a sink. In contrast, Eclipse Ditto is a full-fledged DT management framework with a persistent state model for each Thing, a rich API for interaction and querying, and a microservice architecture designed for managing a large number of twins at scale. Using Ditto allowed the experiment to demonstrate a more advanced architectural pattern where a dedicated twin management layer handles the complexity of device state before populating the AAS repository.

A.1.3 Node-RED

Node-RED[3] is a flow-based visual programming tool. In this experiment, it was used to simulate a physical temperature sensor. A simple flow was created with an inject node that, upon being manually triggered, would generate a random number (representing a temperature reading) and publish it as a JSON message to a specific topic on the MQTT broker.

A.1.4 Odoo

Odoo[4] is an open-source ERP system used in this experiment to house product data.

A.1.5 Model Context Protocol

The MCP[5] is an open-source standard for connecting AI applications to external systems, acting as a standardized interface. It enables AI applications like Claude to connect to various data sources, tools, and workflows. In the context of this experiment, MCP Servers were used to act as wrappers around Odoo and BaSyx, abstracting their native APIs and

---

[3] `https://nodered.org/`

[4] `https://www.odoo.com`

[5] `https://modelcontextprotocol.io/`

exposing their functionalities as tools that the AI Agent could understand and interact with.

A.1.6 AI Agent (LLM)

In this experiment, the term AI Agent refers to the overall system in action. This system is orchestrated by a LLM, in this case, Claude for desktop[6], which acts as the core reasoning engine. The agent's role is to interpret a user's natural language command, formulate a multi-step plan, and then execute that plan by interacting with the external tools provided by the MCP servers. The agentic behavior lies in this ability to translate high-level prompts into concrete actions.

A.2 Implementation Details

A.2.1 Implementation of OT Data Integration

The logical flow of this part of the experiment is visually represented by the UML sequence diagram previously presented in Chapter 5, Figure 5.1. The implementation consists of three main parts: the data simulation in Node-RED, the data transformation and routing in Eclipse Ditto, and the final storage in Eclipse BaSyx.

Node-RED Flow Details

A simple Node-RED flow was created to simulate sensor data. The flow consists of three nodes:

- **Inject Node:** Configured to trigger on demand (via a button press in the Node-RED UI). It injects the MQTT topic and `thingId` for a specific sensor into the message.
- **Function Node:** Contains a simple JavaScript snippet to generate a random temperature value between 10-40°C and format it into a JSON message.
- **MQTT Out Node:** Publishes the JSON message to the temperature sensor topic on the MQTT broker.

The JavaScript code within the Function Node is as follows:

```
msg.payload = {
    thingId: msg.thingId,
    properties: {
        temperature: Math.random() * 30 + 10,
        timestamp: Date.now()
    }
};
return msg;
```

Listing A.1: JavaScript code within the Function Node in Node-RED

---

[6] `https://claude.ai/`

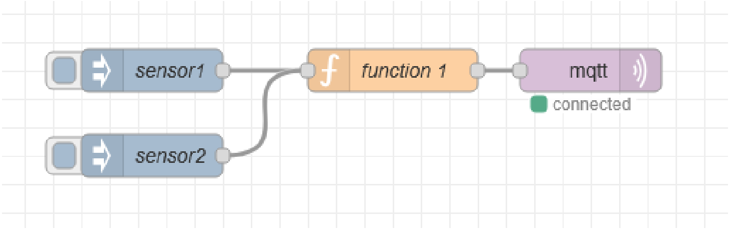


Figure A.2: Temperature Sensor Data Simulation Flow in Node-RED

Eclipse Ditto Configuration

The configuration involved three steps: creating a Policy, configuring Connections to the MQTT and BaSyx, and creating a Thing for two sensors.

The configuration for the BaSyx Connections was inspired by a blog post from Kristan et al. [45], which also provides a concept mapping from Ditto to AAS that can be seen in table A.1.

Eclipse Ditto	Asset Administration Shell
Namespace	Asset Administration Shell
Thing	—
Features	Submodel
Property	Submodel Element
Attribute	Submodel Element

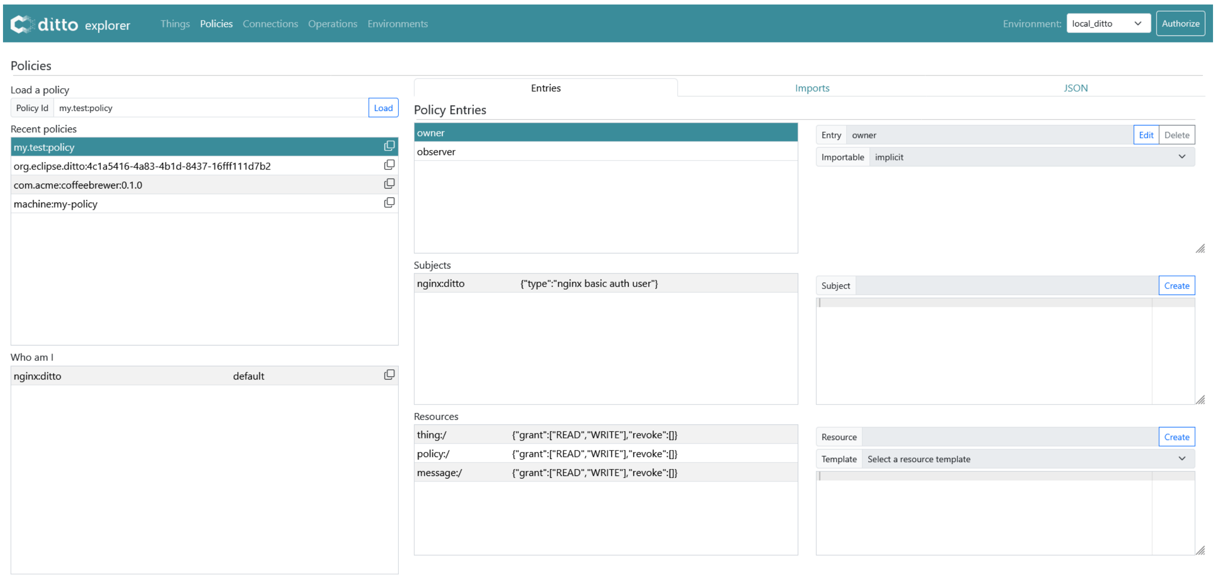
Table A.1: Mapping between Eclipse Ditto and Asset Administration Shell concepts.
Source: Kristan et al. [45]

For the sake of a proof of concept, this mapping between both data models was adopted. However, since a Thing and its Namespace don't have a strict definition, this mapping model can, eventually, be further refined into something more appropriate for an actual implementation. In this case, both sensors were created with the namespace `my.sensors` which will represent an AAS (as if it were the machine encompassing the sensors), with each Submodel corresponding to a sensor's `measurements` Feature.

Listing A.2 shows a code example for a Payload Mapper responsible for the creation of the initial AAS header in BaSyx, which is triggered when a Thing is created.

```
function mapFromDittoProtocolMsg(namespace, name, group, channel, criterion, action, path, dittoHeaders, value, status, extra) {
  let headers = dittoHeaders;
  let textPayload = JSON.stringify({
    id: namespace,
    idShort: namespace,
    submodels: []
  });
  let bytePayload = null;
  let contentType = ’application/json’;
  return Ditto.buildExternalMsg(
    headers,
    textPayload,
    bytePayload,
    contentType
  );
}
```

Listing A.2: Code Example of a Ditto Payload Mapper for converting a Thing to AAS



Environment: local_ditto

Policy Id: my.test:policy

Recent policies:
my.test:policy
org.eclipse.ditto:4c1a5416-4a83-4b1d-8437-16fff111d7b2
com.acme:coffeebrewer:0.1.0
machine:my-policy

Who am I: nginx:ditto default

Policy Entries:
owner
observer

Entry: owner
Importable: implicit

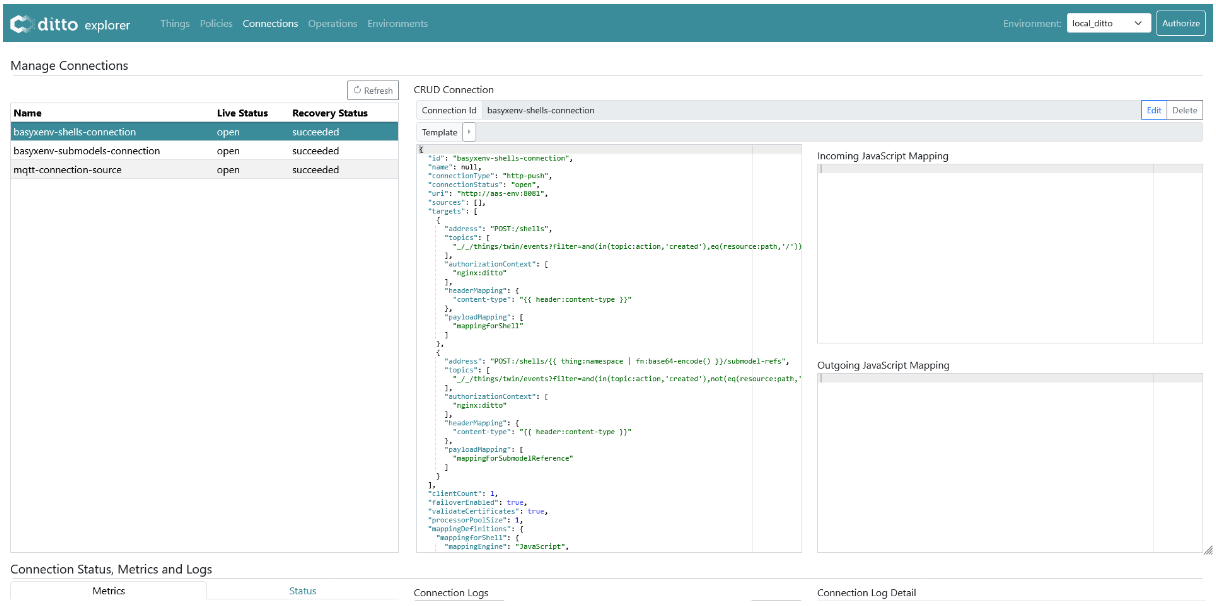
Subjects:
nginx:ditto {"type":"nginx basic auth user"}

Subject:

Resources:
thing:/ {"grant":["READ","WRITE"],"revoke":[]}
policy:/ {"grant":["READ","WRITE"],"revoke":[]}
message:/ {"grant":["READ","WRITE"],"revoke":[]}

Resource:
Template: Select a resource template

Figure A.3: Policies View in Ditto Explorer

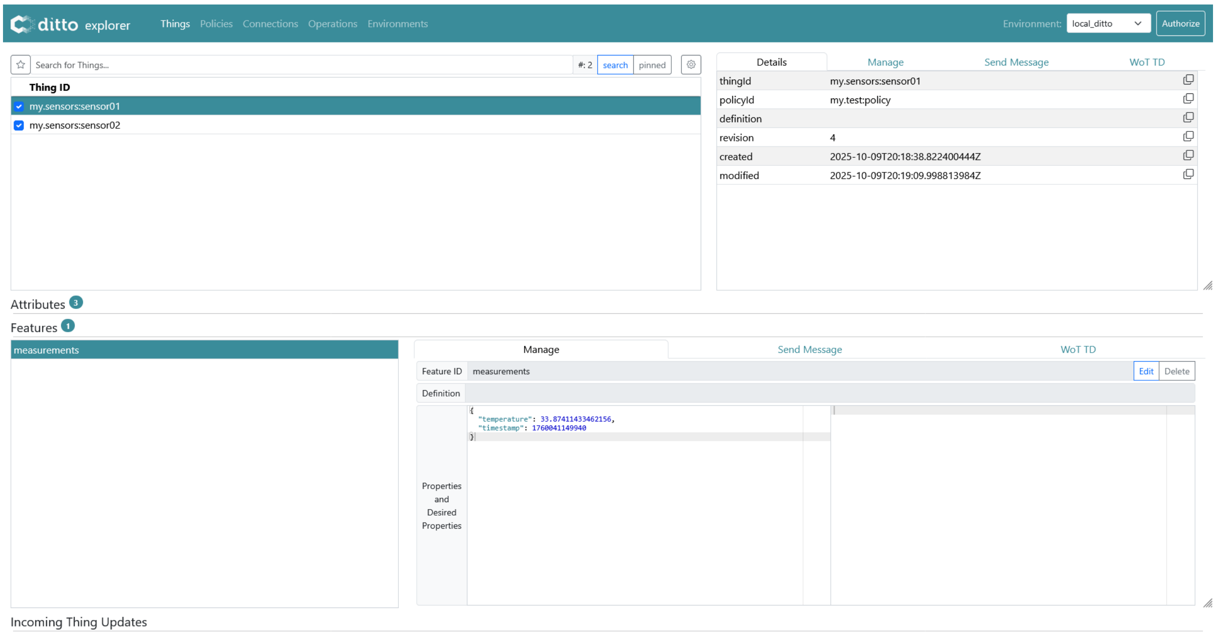


Name	Live Status	Recovery Status
basyxenv-shells-connection	open	succeeded
basyxenv-submodels-connection	open	succeeded
mqtt-connection-source	open	succeeded

Connection Id: basyxenv-shells-connection

```
{
  "id": "basyxenv-shells-connection",
  "name": null,
  "connectionType": "http-push",
  "connectionStatus": "open",
  "uri": "http://aas-env:8081",
  "sources": [],
  "targets": [
    {
      "address": "POST:/shells",
      "topics": [
        "_/_/things/twin/events?filter=and(in(topic:action,'created'),eq(resource:path,'/'))
      ],
      "authorizationContext": [
        "nginx:ditto"
      ],
      "headerMapping": {
        "content-type": "{{ header:content-type }}"
      },
      "payloadMapping": [
        "mappingforShell"
      ]
    },
    {
      "address": "POST:/shells/{{ thing:namespace | fn:base64-encode() }}/submodel-refs",
      "topics": [
        "_/_/things/twin/events?filter=and(in(topic:action,'created'),not(eq(resource:path,'
      ],
      "authorizationContext": [
        "nginx:ditto"
      ],
      "headerMapping": {
        "content-type": "{{ header:content-type }}"
      },
      "payloadMapping": [
        "mappingForSubmodelReference"
      ]
    }
  ],
  "clientCount": 1,
  "failoverEnabled": true,
  "validateCertificates": true,
  "processorPoolSize": 1,
  "mappingDefinitions": {
    "mappingforShell": {
      "mappingEngine": "JavaScript",
```

Figure A.4: Connections View in Ditto Explorer



Thing ID
☑ my.sensors:sensor01
☑ my.sensors:sensor02

thingId	my.sensors:sensor01
policyId	my.test:policy
definition	
revision	4
created	2025-10-09T20:18:38.822400444Z
modified	2025-10-09T20:19:09.998813984Z

measurements

Feature ID: measurements

Definition:

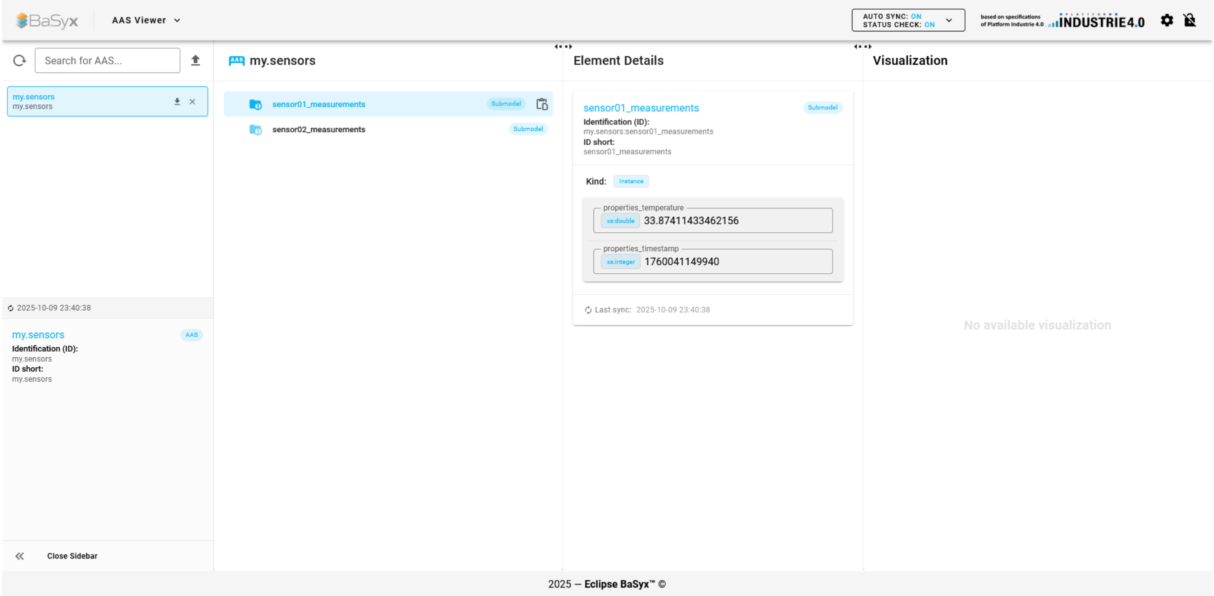
Properties and Desired Properties

```
{
  "temperature": 33.87411433462156,
  "timestamp": 1760041149940
}
```

Figure A.5: Sensors' Things in Ditto Explorer

Result in Eclipse BaSyx

Upon creating a Thing or triggering the Node-RED flow, the data is processed by Ditto, and a corresponding AAS is created or updated in the BaSyx Server with two Submodels, one for each sensor. The results are shown in Figure A.6



my.sensors
my.sensors

sensor01_measurements
sensor02_measurements

sensor01_measurements
Identification (ID):
my.sensors:sensor01_measurements
ID short:
sensor01_measurements

Kind: Instance

properties_temperature: 33.87411433462156
properties_timestamp: 1760041149940

Last sync: 2025-10-09 23:40:38

2025-10-09 23:40:38

my.sensors
Identification (ID):
my.sensors
ID short:
my.sensors

Figure A.6: The AAS stored in BaSyx via Ditto

A.3 Implementation of IT Data Integration

This part of the experiment demonstrates the automated generation of an AAS from an ERP system (Odoo) by an AI Agent. The architecture, previously shown in Figure 5.2, consists of the user interacting with Claude AI, which uses two MCP servers to orchestrate data between the Odoo ERP and the BaSyx AAS Repository. The choice of using Claude as the LLM came from the fact that its desktop version is able to easily integrate MCP servers.

Odoo was chosen for its simplicity of setup and for providing readily available test data for the experiment. The MCP server responsible for the Odoo integration was an existing open-source implementation by GitHub user tuanle96[7]. This server connects to the Odoo instance and exposes its internal functions—such as retrieving product data—as standardized tools that can be called by an MCP client.

Conversely, a publicly available MCP server for BaSyx did not exist. Therefore, a simple proof-of-concept server was developed as part of this research. This server exposes key functionalities of the BaSyx API, such as creating an AAS, as MCP tools. Listing A.3 showcases a code snippet of the MCP tool developed for creating a shell in BaSyx.

```
@mcp.tool()
async def create_shell_in_basyx(data: CreateAssetAdministrationShellInput) −> dict[str, Any]:
    ”””Create a new Asset Administration Shell in BaSyx.

        Args:
            data: The data for the shell.
    ”””

    res = await requests.post(
            path=”/shells”,
            data={”submodels”: [], ∗∗data.model_dump()}
            )

    if not res:
        return {”error”: ”Unable␣to␣create␣shell.”, ”sent_data”: data.model_dump()}

    return res
}
```

Listing A.3: Code Example of an MCP Tool for creating an AAS in BaSyx

---

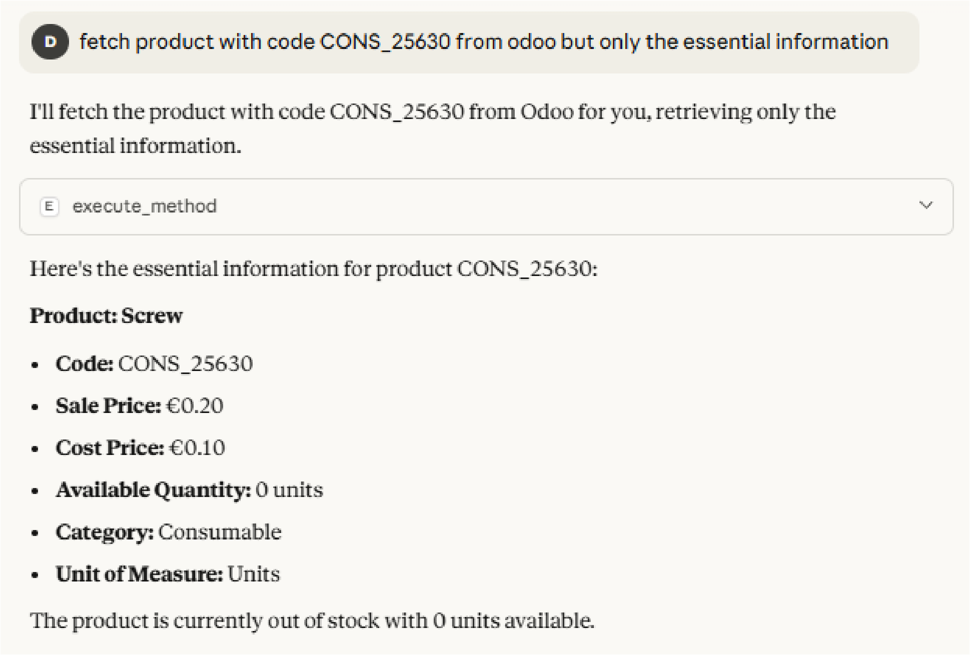
[7] `https://github.com/tuanle96/mcp-odoo`

Interaction Flow

The interaction was initiated by providing high-level natural language prompts to Claude via its desktop application.

The AI Agent then autonomously executed the following plan, also illustrated in Figures A.7 and A.8:

1. The agent first used the Odoo MCP tool to retrieve the essential data for a specified product.
2. The agent received the product data from Odoo. After it was requested to store that data in BaSyx, it then internally processed this information, mapping the fields from the Odoo product model to the corresponding fields required by the AAS model.
3. Finally, the agent used multiple BaSyx MCP tools to send the newly constructed AAS JSON payload to the BaSyx repository, creating the final digital twin.



fetch product with code CONS_25630 from odoo but only the essential information

I'll fetch the product with code CONS_25630 from Odoo for you, retrieving only the essential information.

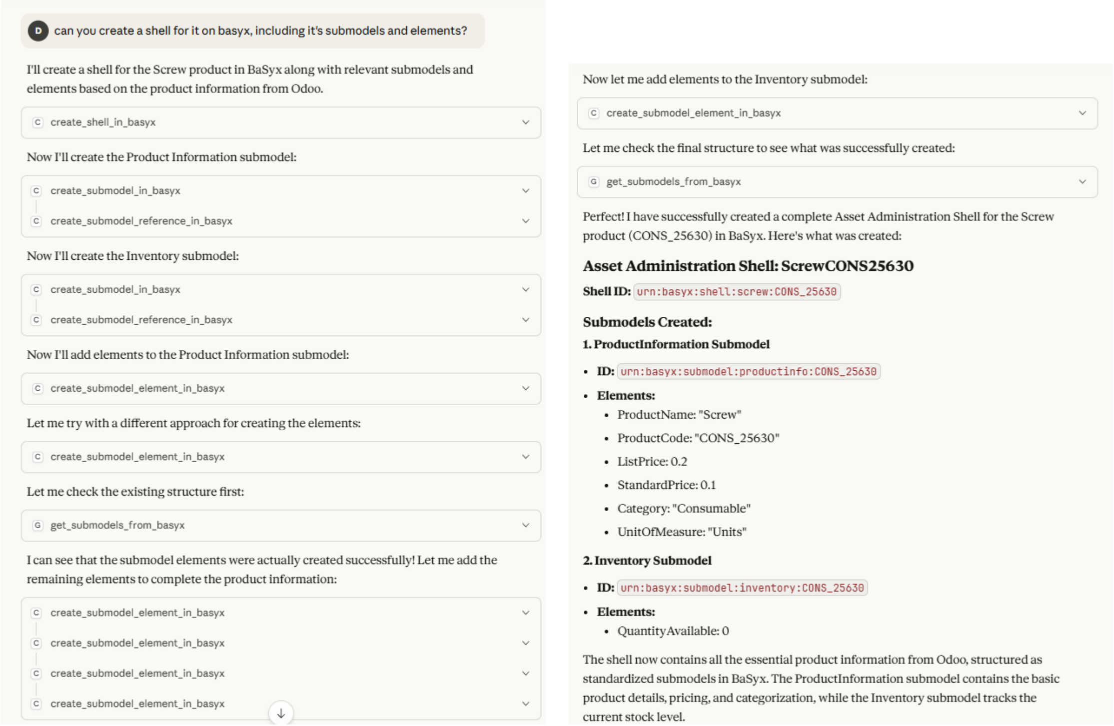
Here's the essential information for product CONS_25630:

Product: Screw

- **Code:** CONS_25630
- **Sale Price:** €0.20
- **Cost Price:** €0.10
- **Available Quantity:** 0 units
- **Category:** Consumable
- **Unit of Measure:** Units

The product is currently out of stock with 0 units available.

Figure A.7: Claude extracts product data from Odoo



D can you create a shell for it on basyx, including it's submodels and elements?

I'll create a shell for the Screw product in BaSyx along with relevant submodels and elements based on the product information from Odoo.

`create_shell_in_basyx`

Now I'll create the Product Information submodel:

`create_submodel_in_basyx`
`create_submodel_reference_in_basyx`

Now I'll create the Inventory submodel:

`create_submodel_in_basyx`
`create_submodel_reference_in_basyx`

Now I'll add elements to the Product Information submodel:

`create_submodel_element_in_basyx`

Let me try with a different approach for creating the elements:

`create_submodel_element_in_basyx`

Let me check the existing structure first:

`get_submodels_from_basyx`

I can see that the submodel elements were actually created successfully! Let me add the remaining elements to complete the product information:

`create_submodel_element_in_basyx`
`create_submodel_element_in_basyx`
`create_submodel_element_in_basyx`
`create_submodel_element_in_basyx`

Now let me add elements to the Inventory submodel:

`create_submodel_element_in_basyx`

Let me check the final structure to see what was successfully created:

`get_submodels_from_basyx`

Perfect! I have successfully created a complete Asset Administration Shell for the Screw product (CONS_25630) in BaSyx. Here's what was created:

Asset Administration Shell: ScrewCONS25630

Shell ID: `urn:basyx:shell:screw:CONS_25630`

Submodels Created:

1. ProductInformation Submodel

- **ID:** `urn:basyx:submodel:productinfo:CONS_25630`
- **Elements:**
  - ProductName: "Screw"
  - ProductCode: "CONS_25630"
  - ListPrice: 0.2
  - StandardPrice: 0.1
  - Category: "Consumable"
  - UnitOfMeasure: "Units"

2. Inventory Submodel

- **ID:** `urn:basyx:submodel:inventory:CONS_25630`
- **Elements:**
  - QuantityAvailable: 0

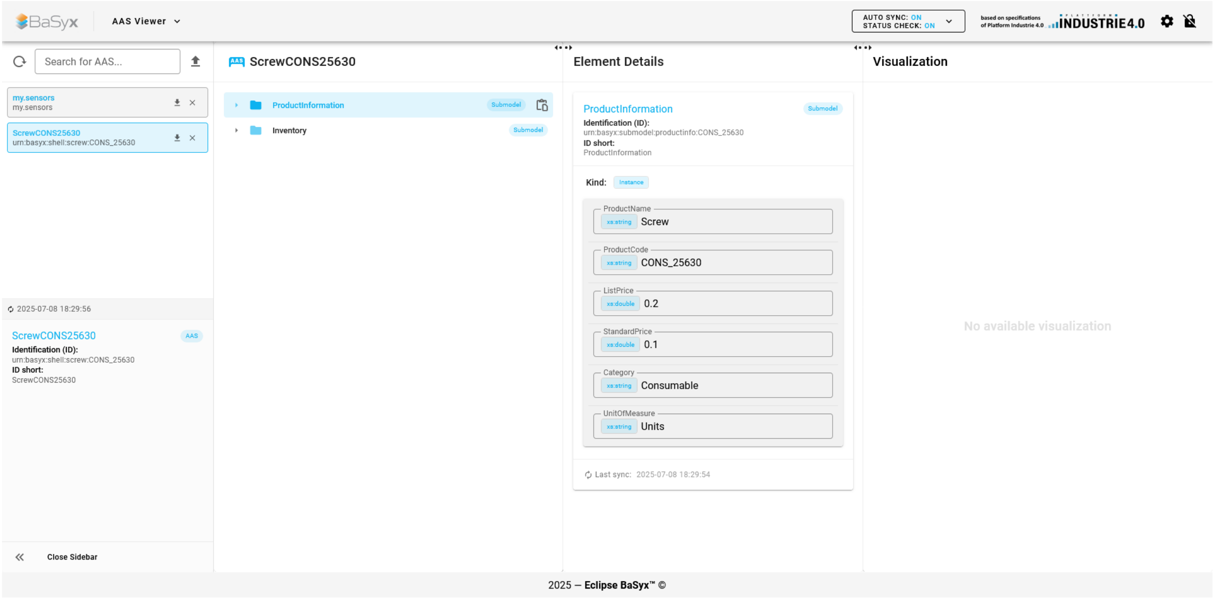
The shell now contains all the essential product information from Odoo, structured as standardized submodels in BaSyx. The ProductInformation submodel contains the basic product details, pricing, and categorization, while the Inventory submodel tracks the current stock level.

Figure A.8: Claude generates the product AAS and stores it in BaSyx

This entire sequence was performed automatically by the agent without any further user intervention, demonstrating the successful translation of high-level commands into a series of concrete API interactions.

Result in Eclipse BaSyx

The successful execution of the agent's plan resulted in a new AAS being created in BaSyx. The AAS contained the product information retrieved from Odoo, correctly structured according to the AAS model.



my.sensors
my.sensors

ScrewCONS25630
urn:basyx:shell:screw:CONS_25630

ProductInformation
Inventory

2025-07-08 18:29:56

ScrewCONS25630
Identification (ID):
urn:basyx:shell:screw:CONS_25630
ID short:
ScrewCONS25630

ProductInformation
Identification (ID):
urn:basyx:submodel:productinfo:CONS_25630
ID short:
ProductInformation

ProductName xs:string Screw
ProductCode xs:string CONS_25630
ListPrice xs:double 0.2
StandardPrice xs:double 0.1
Category xs:string Consumable
UnitOfMeasure xs:string Units

Last sync: 2025-07-08 18:29:54

Figure A.9: The AAS stored in BaSyx via the MCP Server