IOWN GLOBAL FORUM ™

Digital Twin Framework Analysis Report

Classification: REFERENCE DOCUMENT

Confidentiality: PUBLIC

Version 1

[DTF AR]

February 2023

Legal

THIS DOCUMENT HAS BEEN DESIGNATED BY THE INNOVATIVE OPTICAL AND WIRELESS NETWORK GLOBAL FORUM, INC. (“IOWN GLOBAL FORUM”) AS AN APPROVED REFERENCE DOCUMENT AS SUCH TERM IS USED IN THE IOWN GLOBAL FORUM INTELLECTUAL PROPERTY RIGHTS POLICY (THIS “REFERENCE DOCUMENT”).

THIS REFERENCE DOCUMENT IS PROVIDED ”AS IS” WITH NO WARRANTIES WHATSOEVER, WHETHER EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE, INCLUDING WITHOUT LIMITATION ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT OF THIRD PARTY RIGHTS, TITLE, VALIDITY OF RIGHTS IN, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, REFERENCE DOCUMENT, SAMPLE, OR LAW. WITHOUT LIMITATION, IOWN GLOBAL FORUM DISCLAIMS ALL LIABILITY, INCLUDING WITHOUT LIMITATION LIABILITY FOR INFRINGEMENT OF ANY PROPRIETARY RIGHTS AND PRODUCTS LIABILITY, RELATING TO USE OF THE INFORMATION IN THIS REFERENCE DOCUMENT AND TO ANY USE OF THIS REFERENCE DOCUMENT IN CONNECTION WITH THE DEVELOPMENT OF ANY PRODUCT OR SERVICE, AND IOWN GLOBAL FORUM DISCLAIMS ALL LIABILITY FOR COST OF PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, PUNITIVE, EXEMPLARY, OR SPECIAL DAMAGES, WHETHER UNDER CONTRACT, TORT, WARRANTY OR OTHERWISE, ARISING IN ANY WAY OUT OF USE OR RELIANCE UPON THIS REFERENCE DOCUMENT OR ANY INFORMATION HEREIN.

EXCEPT AS EXPRESSLY SET FORTH IN THE PARAGRAPH DIRECTLY BELOW, NO LICENSE IS GRANTED HEREIN, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS OF THE IOWN GLOBAL FORUM, ANY IOWN GLOBAL FORUM MEMBER OR ANY AFFILIATE OF ANY IOWN GLOBAL FORUM MEMBER. EXCEPT AS EXPRESSLY SET FORTH IN THE PARAGRAPH DIRECTLY BELOW, ALL RIGHTS IN THIS REFERENCE DOCUMENT ARE RESERVED.

A limited, non-exclusive, non-transferable, non-assignable, non-sublicensable license is hereby granted by IOWN Global Forum to you to copy, reproduce, and use this Reference Document for internal use only. You must retain this page and all proprietary rights notices in all copies you make of this Reference Document under this license grant.

THIS DOCUMENT IS AN APPROVED REFERENCE DOCUMENT AND IS SUBJECT TO THE REFERENCE DOCUMENT LICENSING COMMITMENTS OF THE MEMBERS OF THE IOWN GLOBAL FORUM PURSUANT TO THE IOWN GLOBAL FORUM INTELLECTUAL PROPERTY RIGHTS POLICY. A COPY OF THE IOWN GLOBAL FORUM INTELLECTUAL PROPERTY RIGHTS POLICY CAN BE OBTAINED BY COMPLETING THE FORM AT: www.iowngf.org/join-forum. USE OF THIS REFERENCE DOCUMENT IS SUBJECT TO THE LIMITED INTERNAL-USE ONLY LICENSE GRANTED ABOVE. IF YOU WOULD LIKE TO REQUEST A COPYRIGHT LICENSE THAT IS DIFFERENT FROM THE ONE GRANTED ABOVE (SUCH AS, BUT NOT LIMITED TO, A LICENSE TO TRANSLATE THIS REFERENCE DOCUMENT INTO ANOTHER LANGUAGE), PLEASE CONTACT US BY COMPLETING THE FORM AT: https://iowngf.org/contact-us/

Contents

List of Figures

List of Tables

Executive Summary

This document provides an analysis for understanding the use cases of digital twins defined by the IOWN Global Forum. The analysis includes the life cycle of digital twins, stakeholders, patterns, data structure and data flow, and technical requirements. This document also summarizes relevant technologies related to digital twin standardization.

From the analysis of five targeted use cases: Area Management Security, Green Twin, Human Digital Twin, Remote Robot Operation, and Area Management Disaster Notification, this document clarifies the three important points to building digital twins: definition of digital twin composition, mechanism for cross-domain data flows, and volume of data.

The four requirements extracted from the use case analysis are compared with the existing technologies to find technical gaps. We identified 7 gaps through the gap analysis, and the technical solutions will be discussed in the Digital Twins Framework Task Force. The IOWN GF will target to resolve these technical gaps by combining relevant technologies to realize the use cases defined by the IOWN Global Forum.

In addition, the technologies related to digital twins are briefly summarized in the Appendix. These relevant technologies will be considered as the foundation of a platform for a digital twin application.

1. Introduction

Digital Twins are considered for visualizing, digitalization, simulation, estimation, and orchestration/control in certain fields such as smart cities, factories, network systems, and so on. In the use case documents of IOWN Global Forum (IOWN GF) [IOWN GF AIC] [IOWN GF CPS], some use cases, such as entertainment and area management security, would utilize digital twins. The objectives of these digital twins might differ from others, so it requires clarifying the usages of these digital twins for establishing the framework of digital twins in IOWN GF.

This document describes the use case analysis of IOWN GF use cases. This analysis aims to clarify patterns of digital twin use, relevant stakeholders, and cross-domain data flow of digital twins. These insights also clarify technical issues to be addressed in the IOWN GF. This report analyzes existing use cases and their Reference Implementation Models (RIMs) defined in the IOWN GF. Based on actual use cases discussed in the IOWN GF documents, concrete requirements and issues will be clarified.

2. Target Use Cases

2.1. Area Management Security

Area Management Security (AMS) is one of the smart city use cases described in [IOWN GF CPS], which aims to keep human safety in nations, cities, buildings, and specific areas. Digital twin applications for AMS may make short-term predictions and generate some proactive actions in order to secure human lives. An AMS digital twin consists of static/semi-static parts, such as building, area map, and environment, and moving/dynamic object parts, such as humans, objects, and security officers. The analyzing result of this RIM indicates the data volumes targeted are estimated as 1.6TB for static/semi-static part, and 9.22TB for moving/dynamic part for 10km x 10km area.

The analysis of AMS digital twin for an airport clarifies necessary data structure and data flows based on several estimations. The results include that the AMS digital twin for an airport digital twin requires 80MB of data volume as the static/semi-static part, 15.66MB for suspicious objects, and 921.6MB for object status and object position data as the moving/dynamic object part. A digital twin of a suspicious object will be used by security officers and airport operators to keep human safety.

2.2. Green Twin: Green Management of Urban Area

Green Twin use case encompasses applications that monitor and coordinate systems' operation and people's activities to reduce energy consumption to help tackle climate change challenges while enhancing their quality of life. Initially, the Green Twin relates to urban areas with three major digital twin categories: building twin, vehicle twin, and person twin. These twin entities can be enhanced with other relevant categories in the future such as forest twin. Data models of Green Twin would span both static (e.g., building 3D models, vehicle consumption model, HVAC datasheets) and real-time or historical streaming (e.g., smart meters, wearable, in-vehicle sensors) data models. Also, a digital twin of network infrastructures is an important entity to manage entire system to keep updating the necessary static and real-time data models [IOWN GF CPS]. The data is used to have a digital twin representation of the real objects that are related to each other through certain links (e.g., an individual is within a building). The data is, then, processed by data analytics operations to infer current status, predict future status or simulate hypothetical conditions. The results of those analytics are used to enhance the data of the digital twin. Stakeholders of this use case lie in two logical categories: Digital Twin users (e.g., asset managers, mobility providers, energy providers), and Digital Twin framework providers (e.g., data providers, analytics model providers, service orchestration providers, data and computing infrastructure providers).

The initial study about the amount of generated data results in setting the needed bandwidth (for a single building and surrounding) of ~80Gbps. We refer to a university building of the Campus of Murcia [UniMurcia Facultad Medicina] composed of 6 levels (2 of which are underground), 500 rooms, and 40 hallways. Detailed analysis on the data volume and velocity of the Green Twin is included in this report.

2.3. Human Digital Twin

A Human Digital Twin (HDT) is a dynamic digital representation of whole or targeted body parts/areas of a physical person. HDT dynamically captures and monitors a human body change and body health condition through imaging scanning data, wearable devices, surrounding sensors, medical testing data, and other inputs.

An HDT helps the medical care teams and individuals to prevent and react earlier and cheaper to current and future medical problems.

An aggregated abstract digital health model can be established from individual DT models with similar demographic characteristics in a region. Any deviation from a normal digital health model may give an early warning of abnormal societal health condition changes such as an emerging pandemic.

Privacy, security, and social ethics must be considered when working with HDT.

2.4. Remote Robot Operation

Remote Robot Operation (RRO), which is the one part of the factory remote operation use cases described in [IOWN GF CPS], is expected to be widely used in near future because of the lack of manpower in an aging society. An example of RRO is remote plant maintenance, that is, on-site robots controlled by a maintenance expert at the remote site can perform necessary maintenance procedures, as if the expert were at the plant site. Digital twin applications for RRO aim to be used for simulation by changing operating parameters, e.g., raw material, reaction, atmospheric condition, or switching production. An RRO digital twin consists of static/semi-static parts, such as plant Building Information Model (BIM) and Facility/Asset (static), and moving/dynamic objects parts, such as the robot, Facility/Asset (dynamic), and Remote Operator. According to the estimation of smart factories, the most severe requirements come from input data such as sensor data (up to 526PB for 10 years) and capturing images (up to 303PB for 10 years).

2.5. Area Management Disaster Notification

Area Management Disaster Notification is one of the smart city use cases described in [IOWN GF CPS]. The goal of the Disaster Notification use case is to alert the people fast and guide escape route correctly when the disaster happened. The Disaster Notification digital twin analysis adopts Taipei city (271.8 km^2), Taiwan as a target area. It consists of static/semi-static parts, such as building, sensing facility, area map, and environment, and dynamic object parts, such as people, objects, and emergency unit. The analyzing result indicates the data volumes are estimated as static resources 137TB and time-series 7TB/s for static/semi-static part; time-series 62.762TB/s for moving/dynamic part.

3. Analysis Results

3.1. Lifecycle of digital twins

According to use case analysis above, the lifecycle of digital twin is identified as shown in Figure 3.1-1.

The lifecycle of digital twins has four stages; create, update/customize, consume, and delete. Create stage means that an owner creates a digital twin based on digitalized thing or concept (e.g., process, value, or money) through pre-defined data structure. Update/Customize stage means that the created digital twin is customized for specific purpose and condition and is continuously updated with the latest state and position. Consume stage is used by a consumer of the digital twin. In this stage, a digital twin might be used for many types of consumers from different companies, domains or industry in typical use cases. In addition, Update/Customize and Consume may occur in parallel if needed. Finally, at Delete stage, the digital twin is disposed by an owner of the digital twin.

In addition, there is a complex case for Create or Update/Customize, as shown in the left side of Figure 3.1-1. If a created digital twin has a large and complex structure/system, it could be a collection of specific parts of digital twins, called primitive digital twins, in some use cases (e.g., airport, factory, city, and network infrastructures). In this case, there are additional processes to create primitive digital twin before Create and Update/Customize stages.

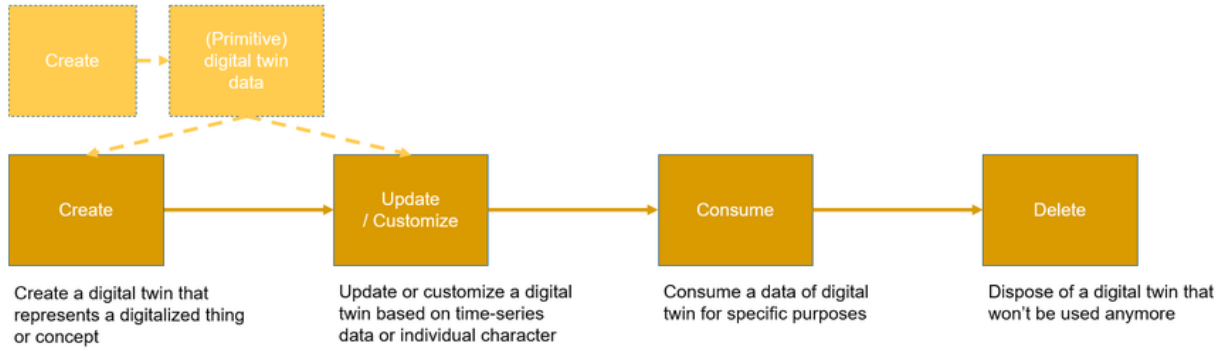

Figure 3.1-1: Lifecycle of digital twins

3.2. Analysis of stakeholders

In order to clarify relevant stakeholders and roles, stakeholders are analyzed. The results observe that the following stakeholders who have specific roles in the digital twin lifecycle.

Different stakeholders and their behaviors have been identified in each stage of the lifecycle, depending on use cases. The basic pattern is that a use case for an open area where an indefinite number of people exists (e.g., Area Management Security, and Green Twin) has relatively many stakeholders, especially for Create and Consume roles, while a use case for a closed area (e.g., Remote Robot Operation in a factory) has a relatively small number of stakeholders and a limited number of Creators and Consumers.

Regarding the open area use cases, cross-domain / company data flow between different stakeholders exists. For example, the digital twin of the facility is created by a real estate developer, then the digital twin is consumed by a government, a security company, or a resident. This pattern is also observed in the Green Twin use case. On the other hand, the Remote Robot Operation use case has relatively limited number of stakeholders who belong to the same industry. The reason is that the digital twin is used for sharing data with a definite number of stakeholders as the use case focuses on a specific purpose in a specific facility.

Thus, there are at least two types of use cases, one is an open area use case where many types of stakeholders exchange data beyond a company, a domain, and an industry, and another one is a closed area use case where a limited number of stakeholders create and consume data in the same supply chain.

Number of types/categories of stakeholders is also closely related to privacy and security concerns and may be kept under control. In addition, identity of a HDT subject may not be exchanged among different stakeholders, even though non-personal information may be shared.

3.2.1. Area Management Security

The following Table 3.2-1 shows stakeholder analysis results for Area Management Security.

Table 3.2-1: Stakeholders of Area Management Security

Stakeholders	Stakeholder Role	Action
DT Platform Operator	A platform provider to create and operate a virtual space where digital twins run.	Create Update / Maintain Delete
AMS Application Provider	An application provider for the Area Management Security.	Update / Maintain Consume
Map Company	A company providing a digital map.	Create Update / Maintain Delete
Real Estate Developer **(Model provider of a real estate)**	A developer of a real estate providing a model of facility.	Create Delete
Manufacturer **(Model provider of a product)**	A manufacturer of a product providing a model of physical product.	Create Delete
Model Provider of human digital twin	A digital data creator providing a model of human digital twin.	Create Delete
Asset Owner / Asset Agent	An owner or an agent of an asset / a real estate.	Update / Maintain
Government (Police)	Division of government (in this case, police).	Consume
Security Company	A company providing security services.	Consume
Tenant	A tenant of an asset / a building.	Consume
Resident	A resident of an asset / a building.	Consume

3.2.2. Green Twin

The following Table 3.2-2 shows stakeholder analysis results for Green Twin.

Table 3.2-2: Stakeholders of Green Twin

Stakeholders	Stakeholder Role	Action
DT Platform Operator	A platform provider to create and host the digital twin based on standard data model and run digital twin services (e.g., compute services or simulations).	Create Update / Maintain Delete
Model Provider (Green Twin)	A digital data creator providing a model of digital twin using standardized data models.	Create Update Delete
Digital Twin Application Provider	The digital twin application provider based on the relevant customer stakeholders.	Update / Maintain Create actionable insights
Asset Owner / Management	An owner or manager of building(s) or vehicle(s).	Consume Make/Apply decision on asset management
Mobility Service Providers/Management	The providers of the mobility services for the region.	Consume Make/Apply decision on asset management
Individual Users	Person(s) that use digital twin services for reducing energy consumption and/or improving quality of life (e.g., safer driving, eco alternative to cars, comfort into buildings).	Consume Make/Apply decision on asset management
Policymakers	Policymakers of the region for energy consumption and costs.	Consume Make/Apply decision on asset management
Vehicle Manufacturer	Manufacturer of vehicles.	Create Update Delete
Energy Provider	Manufacturer and provider of (renewable) energy sources	Create Update / Maintain Delete
Real Estate Developer	Building developer for smart buildings with energy management.	Create Update / Maintain Delete
Data Providers	Open or proprietary data providers	Provide Data Update / Maintain

3.2.3. Remote Robot Operation

The following Table 3.2-3 is the result of stakeholder analysis for Remote Robot Operation.

Table 3.2-3: Stakeholders of Remote Robot Operation

Stakeholders	Stakeholder Role	Action
DT Platform Operator	A platform provider to create and operate a virtual space *1 where digital twins run.	Create Update / Maintain Delete
RM Application Provider	An application provider for the Robot Management.	Update / Maintain Consume
Factory / Facility Owner	A company owns factory, facility and their digital information including a map.	Create Update / Maintain Delete
Real Estate Developer	A developer of a factory and facility.	Create Delete
Manufacturer	A manufacturer of a robot.	Create Consume Delete
Factory Operator	A company operates robots in a factory remotely.	Consume

*1 see “Definitions” section for details.

3.2.4. Area Management Disaster Notification

The following Table 3.2-4 is the result of stakeholder analysis for Area Management Disaster Notification.

Table 3.2-4: Stakeholders of Area Management Disaster Notification

Stakeholders	Explanation	Action
DT Platform Operator	A platform provider to create and operate a virtual space where digital twins run.	Create Update / Maintain Delete
Application Provider	An application provider.	Create Update / Maintain Consume Delete

Map Company	A company providing a digital map. (road, zone, river, terrain, etc.)	Create Update / Maintain Delete
Real Estate Developer	A developer of a real estate. (apartment, public building, etc.)	Create Delete
Asset Manufacturer	A manufacturer of a product providing a model of asset digital twin. (car, MRT etc.)	Create Delete
Model Provider	A digital data creator providing a disaster behaviors model for digital twin.	Create Delete
Asset/Facility Owner	An owner or an agent of an asset and facility.	Create Update / Maintain Consume Delete
Government	Division of government (in this case, police, fire fighter, military).	Create Update / Maintain Consume Delete
People	People in the target area.	Consume / User

3.3. Analysis of patterns of digital twins

Through analysis of use cases, several patterns of digital twins are identified. At each lifecycle stage, several common characters of digital twin patterns can be summarized as follows. These characters are mainly extracted by the Area Management Security, but they should be applied to other use cases as well.

- Create
  - A digital twin of a man-made object has its manufacturer or developer who provides a model of a digital twin.
  - A digital twin of human, or natural environment has no apparent designer or creator. Thus, a model provider is needed for generating these digital twins. A model could be provided by a service platformer or an application provider or could be defined as a standard data model by national/international standardization organizations.
  - A map is created by an apparent developer: a map company and provides common geographic information even if buildings and roads are not owned by the map company. There will be a similar role for other things such as digital twins for environment, society, and economy in the future.
  - Volume of digital twin data has a correlation with scale and complexity of digital twin. In general, a large-scale digital twin or a complex digital twin requires more data volume than simple one.

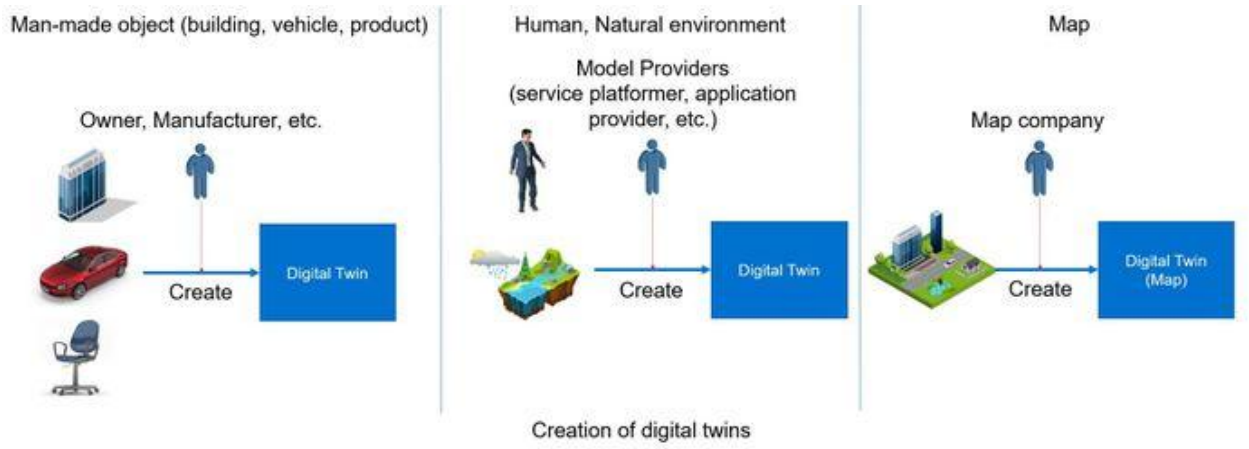

Figure 3.3-1: Creation patterns of Digital Twin

- Update
  - A digital twin needs to be updated for application-specific purpose by application providers or asset owners. Update frequency is vary based on their purpose and nature of dynamic / static.
  - Dynamic digital twin requires frequent updates and stores time series data as its history. These updates and history data generate a lot of data transaction and need huge storage space in general. This is important requirement for use of a dynamic digital twin.
  - On the other hand, static digital twin requires relatively small volume of data transaction to be updated and does not require a lot of time series data to store its history. Even if the digital twin is a large scale of building or facility, the total amount of digital twin data is relatively small.
  - A digital twin, in general, is customized to fit for their purpose by application providers or asset owners. For example, a vehicle digital twin is customized based on their capability, owner, and status to be used for specific purpose and environment.
  - Application providers may merge several digital twins (or sub-parts of digital twins) to create a complex digital twin for a large building, a factory/plant map, and a virtual space.

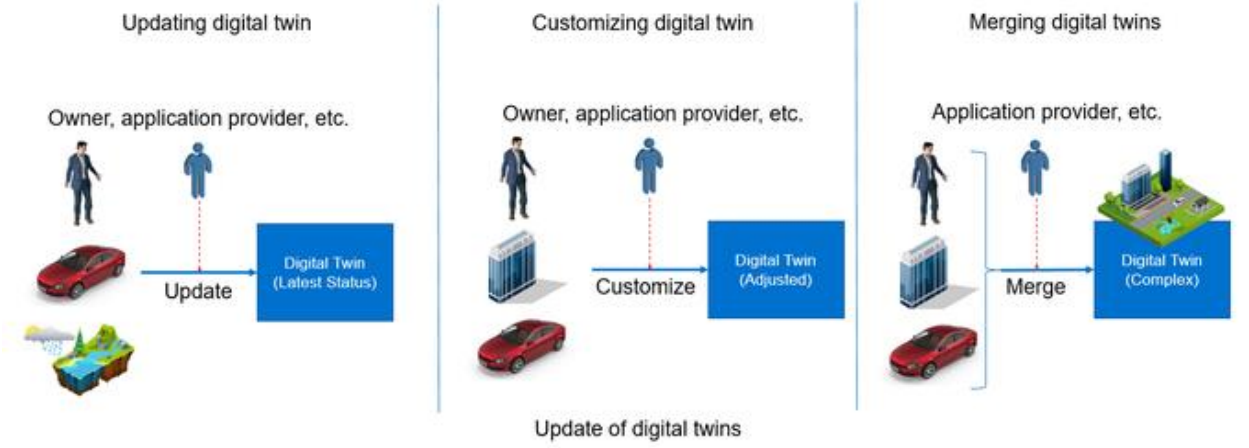

Figure 3.3-2: Update patterns of Digital Twin

- Consume
  - There are many types of consumers of a digital twin in use cases analyzed. For example, they are different domain, company, industry from their creators and operators (updating / customizing).

- The trust level of consumers is not the same. For example, factory owners in the robot management use case must be trusted for consuming and maintaining a digital twin of factory, and they could be a part of an application provider. On the other hand, a resident and a passenger (indefinite number of people) are not fully trusted for digital twins for AMS and Green Twin use case, since they can just get the digital twins without a right to modify the digital twin. Access control and security suitable for trust level are required for consuming digital twins.
- In case of a complex digital twin (e.g., a large building, virtual space), multiple stakeholders collaborate to create a digital twin, and the integrated digital twin is consumed by different stakeholders who are usually from different domains or industries. This situation creates an environment of cross-industry / domain / company sharing of a digital twin.

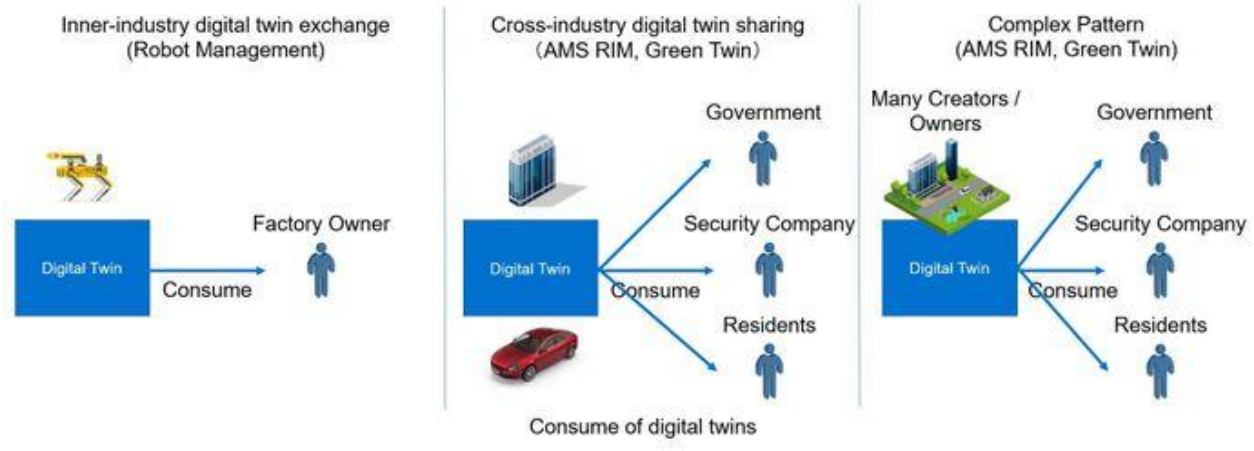

Figure 3.3-3: Consume patterns of Digital Twin

The following tables show the necessary digital twins in the target use-cases with their details.

3.3.1. Area Management Security

Table 3.3-1: Possible Patterns of AMS Digital Twins [Static / Semi-static]

Pattern of Digital Twin	Building / Asset digital twin	Area map	Environment (May not be a digital twin, just an environmental condition of target area)
Outline	Representing structure of a building, road, asset (e.g., security gate)	Representing map of target area	Representing condition of target area (e.g., weather, brightness)
Unit	One building / one asset	Target area	Target area
Necessary Information	● position ● rotation ● geometry ● material (Most information should be pre-defined)	● map information including road, zone, facilities (e.g., police office) (Basically, this is pre-defined)	● weather information (temperature, rain) from external system ● brightness (light, fog) from sensors

Time-series Information	If a digital twin is modified (e.g., broken), time and modification should be reflected.	Basically N/A, unless there is change on map	changes of environmental conditions
Create	Real Estate Developer	Map Company	Service Platformer
Update / Maintenance	Asset Owners / Asset Agents	Map Company	Service Platformer
Consume / User	AMS Application Provider Government (police) Security Company Tenant Resident	AMS Application Provider Government (police) Security Company Tenant Resident	AMS Application Provider Government (police) Security Company Tenant Resident
Data Volume	1.6TB 10000 areas (10km x 10km) / year		

Table 3.3-2: Possible Patterns of AMS Digital Twin [Moving / Dynamic Objects]

Pattern of Digital Twin	Man (e.g., suspect, pedestrian)	Object (e.g., spanner, car)	Officers (e.g., police, security guards)
Outline	Representing man-like moving object including their behavior	Representing labeled object such as vehicles, tools, and weapons	Representing locations of officers to prevent security breach
Unit	Individual	Individual object	Individual
Necessary Information	● Appearance data from Camera, LIDAR ● Behavior data from time-series data analysis ● Position from LIDAR ● Movement path from time-series data analysis)	● Appearance data from Camera, LIDAR ● Behavior / modification data from time-series data analysis ● Position from LIDAR ● Movement path from time-series data analysis	● Position from GPS data ● Status from availability information
Time-series Information	● Movement path ● Behavior data	● Movement path ● Behavior data ● Change of shape / modification	● Movement path?
Create	Model Provider (Human DT)	Manufacturer	Service Platformer
Update / Maintenance	AMS Application Provider	AMS Application Provider	AMS Application Provider
Consume / User	Government (police) Security Company	Government (police) Security Company	Government (police) Security Company

Data Volume	Latest Status: 128GB Time Series: 9.22TB Behavior Pattern: 12.6GB 10000 areas (10km x 10km) / year

3.3.2. Green Twin

Table 3.3-3: Possible pattern of Green Twin

Pattern of Digital Twin	Smart building	Vehicle	Person
Outline	Building model (including floor maps/rooms and sensors)	Vehicular mobility attributes and dynamic real-time information and historical data	Personal mobility behaviors/choices, recommendations, and social information
Unit	Entire building	Vehicle model	Individual model
Necessary Information	● building 2D/3D model ● sensor data ● energy consumption ● HVAC operations ● human activity (presence in the building, scheduled events, type of activities expected) (pre-defined or inspected) Building model would contain the room sizes, floor maps, material type, and other relevant data.	● vehicular sensor data ● position ● velocity ● routes and traffic data ● environment data (inspected) Vehicular sensing data includes the sensing data that goes out of the vehicle, as opposed to the data that is consumed and deleted in-vehicle.	● mobility trajectory ● personal mobility choices ● individual CO2 emission ● personal schedule ● mobility services data (inspected) Privacy-sensitive personal data may reside on the individual's device.
Time-series Information	If the digital twin is modified (e.g., renovation), time and modification should be reflected. Historization of versions of the same Digital Twin. Sensor observations, energy-meters counters, and HVAC operations	Most of the information is time-series due to their nature such as velocity, position, and CO2 emissions, except than static environmental information such as roads	Most of the information is time-series including mobility choices or CO2 emission. If the digital twin is modified (e.g., new mobility services), time and modification should be reflected.
Consumer / User	Building management	Passengers, mobility service provider, remote traffic control center, and vehicle manufacturers	Passengers/pedestrians, mobility service provider and smart city control center
Data Volume	Building model up to TBs Sensor data MBs per second	Vehicle sensor data up to GBs per second	Personal mobility data up to MBs per second Mobility services data up to GBs per second

Benefit	Energy consumption reduction including heating and electric	CO2 emission reduction, time efficiency, and reduced risk of accidents	CO2 emission reduction, time efficiency, personal comfort improvement, and improved city services

3.3.3. Human Digital Twin

Table 3.3-4: Possible Patterns of Digital Twins [Static / Semi-static]

Pattern of Digital Twin	Human digital twin	Target area human body digital model	Note
Outline	Representing digital form of a human body	Representing digital form of a target body parts/areas	Representing condition of a human body (e.g., deviation from normal health condition)
Unit	One person	Target body parts/areas	
Necessary Information	Magnetic resonance imaging. (MRI) whole body scan or other forms of medical images of a human body Real time or high frequent readings from wearable devices which capture data of a human's position, location, heart rate, temperature, blood pressure, oxygen level, blood sugar level, breath et al.	Can be targeted for a certain section of body, e.g., brain, stomach	weather information (temperature, rain, humidity) from external system brightness (light, fog) from sensors environment (e.g., desert, mountain, urban)
Time-series Information	A new MRI scan data or significant change of health indicators will trigger modification of the digital twin. Note that a human digital twin has a time-dependence models with time and modification recorded.	changes in target body parts/areas	changes of health conditions
Place for Processing	In edge nodes or CCC (Central Control Center, e.g., a datacenter of a health care provider network where data are stored and processed), possible in end device when its computation/storage/rendering capabilities are sufficient to support DT	In edge nodes or CCC (Central Control Center), possible in end device when its computation/storage/rendering capabilities are sufficient to support DT	In edge nodes or CCC (Central Control Center), possible in end device when its computation/storage/rendering capabilities are sufficient to support DT
Consumer / User	Individual/Doctors	Individual/Doctors	Individual/Doctors
Data Volume	Depends, most from MRI scan or other form image data, a few Gbyte (GB) to hundreds of GB, up to petabyte (PB)/individual considering historical data and resolution.		
Benefit	Individualized health assessment, advice, early detection, prevention	Individualized health assessment, advice, early detection, prevention	Individualized health assessment, advice, early detection, prevention

3.3.4. Remote Robot Operation

Table 3.3-5: Possible Patterns of Digital Twins [Static / Semi-static]

Pattern of Digital Twin	Plant Structure	Facility / Asset (static)
Outline	Representing structure of a plant (including facilities and assets)	Representing static structure of a building, an asset, an object (e.g., facility, pipe, valve)
Unit	Entire plant	Individual object
Necessary Information	● geometry ● position ● rotation ● material (pre-defined or inspected) (Probably this is a complex of various types of digital twins (individual facility/asset))	● geometry ● position ● rotation ● material (pre-defined or inspected)
Time-series Information	If the digital twin is modified (e.g., broken), time and modification should be reflected.	If the digital twin is modified (e.g., broken) or moving, time and modification should be reflected.
Create	Service Platformer Factory / Facility Owner Real Estate Developer	Service Platformer Factory / Facility Owner
Update / Maintenance	Factory / Facility Owner	Factory / Facility Owner
Consume / User	RM Application Provider	RM Application Provider

Table 3.3-6: Possible Patterns of Digital Twin [Moving / Dynamic Objects]

Pattern of Digital Twin	Robot (e.g., monitoring, delivering, inspecting, maintenance robots)	Facility / Asset (dynamic)	Remote Operator
Outline	Representing location, status and behavior of various type of robots	Representing structure of a building, an asset, an object (e.g., facility, pipe, valve) in real-time (shown as Live Streaming or Hologram)	Representing behavior or movement of remote operator who is using maintenance robot
Unit	Individual robot	Individual object	Individual operator

Necessary Information	• Appearance data (shape, weight, size) • Position / Rotation • Function (monitoring, maintenance) • Haptic information	• Camera Data • Sensor Data • Position / Rotation (monitored or inspected)	• Body tracking • Haptics information (Behaviors to control remote robots are digitalized. This could be partial body tracking data (e.g., arms, fingers), voice or click/pinch a hologram command)
Time-series Information	• Control data • Movement path • Behavior data • Change of shape / modification	If the digital twin is modified (e.g., broken) or moving, time and modification should be reflected.	• Movement • Behavior • Control Command
Create	Manufacturer	Factory / Facility Owner	RM Application Provider
Update / Maintenance	Factory / Facility Owner RM Application Provider	Factory / Facility Owner RM Application Provider	RM Application Provider
Consume / User	Factory Operator	Factory Operator	Factory Operator

3.3.5. Area Management Disaster Notification

Table 3.3-7: Possible Patterns of Digital Twins [Static / Semi-static]

Pattern of Digital Twin	Building / public infrastructure	Sensing facility	Area map
Outline	Building model (including floor maps, rooms, sensors, asset) Public infrastructure (e.g., railway path, park, pipeline, shelters)	Dedicated facility or point which is installed the sensors for detecting the disaster and environment situation (e.g., seismograph, rain gauge, water level)	Geographic data of target area
Unit	One building / one asset	One asset	Target area
Necessary Information	• Building and asset 2D/3D model • position • sensor data *pre-defined or inspected *Building model would contain the room sizes, floor maps, material type, equipment, and other relevant data.	• sensor data • sensor position	• map information including road, zone, river, terrain, mountain, sea, every building and asset (Basically this is pre-defined)

Time-series Information	If real world is changed (e.g., building broken), and changes should be reflected in digital twin. Sensor information is time-series and changed by environmental conditions.	Most of the information is time-series and changed by environmental conditions such as earthquake happened, rain falling etc.	N/A, but map data shall be updated frequently.
Place for Processing	● Control Center in the Building	● regional cloud center	● regional cloud center
Create	● Real Estate Developer	● Asset/Facility Owner	● Map Company
Update / Maintenance	● Asset/Facility Owner	● Asset/Facility Owner	● Map Company
Consumer / User	Application Provider Asset/Facility Owner Government People	Application Provider Government Asset/Facility Owner	Application Provider Government People

Data Volume	BIM Model (LOD400) ● House: 6.817 TB ● Small building: 26.22 TB ● Medium building: 95.25 TB ● Large building: 7.648 TB Total: 135.93 TB Sensor inside building ● House: 409 GB/s ● Small building: 1.573 TB/s ● Medium building: 4.572 TB/s ● Large building: 382.4 GB/s Total: 6.936 TB/s *Number of buildings according to open data by Ministry of Interior Taiwan 2022 *BIM Model LOD400 *Assuming Sensor data: 100KB/s House (<500 m^2): ● 136,349 blocks ● Model size: 50MB/each block ● Assuming number of sensor: 30 Small building (500-5000 m^2): ● 52,447 block ● Model size: 500MB/each block ● Assuming number of sensor: 300 Medium building (5000-20000 m^2): ● 38,102 block ● Model size: 2.5GB/each block ● Assuming Number of sensor: 1200 Large building (>20000m^2): ● 1,912 block ● Model size: 4GB/each block ● Assuming Number of sensor: 2000	Outside Sensor Total: 1 GB/s *Sensor data: 500KB/s *Assuming target area as a center to collect within a 500 km radius and total number of sensor is 2000 pieces.	Base map: 70MB Terrain: 6MB Orthophoto: 1TB 3D Mesh: 3.22TB Total: 4.22 TB *Terrain data: DTM 20M *Orthophoto: GSD 3cm/1px

Table 3.3-8: Possible Patterns of Digital Twin [Moving / Dynamic Objects]

Pattern of Digital Twin	People (e.g., tenant, resident)	Object (e.g., car, train)	Emergency unit (e.g., police, fire fighter, military)
Outline	representing people	representing labeled object such as vehicles, transportation system, etc.	representing locations of emergency unit to deploy fast
Unit	Individual	Individual object	Individual
Necessary Information	● Position from GPS, 4G/5G ● Movement path from time-series data analysis)	● Position from GPS, 4G/5G ● Movement path from time-series data analysis ● Passenger information	● Position from GPS, 4G/5G ● Status from availability information ● Rescuer vital sign
Time-series Information	● Movement path ● position	● Movement path ● Vehicle information (speed, position)	● Movement path ● Position ● Vital sign
Place for Processing	● regional cloud center	● regional cloud center	● regional Cloud center
Create	Application Provider	Asset Manufacturer	Government
Update / Maintenance	Application Provider	Asset/Facility Owner	Government
Consume / User	Government	Government Asset Owner	Government
Data Volume	Total: 26.460 TB/s *Assuming each person: 10MB/s *2.646 million people in Taipei city *This UC's goal is to alert people fast and guide when disaster coming, so that is no need to build human DT detailly.	Total: 35.402 TB/s *Assuming each object: 20MB/s *1.77 million vehicles in Taipei city *120 trains, MRT	Total: 900 GB/s *Assuming each person: 30MB/s *2,000 fire fighter *8,000 police officer *20,000 military and Private rescuer

3.4. Analysis of Data Structure & Flow

A data structure of a digital twin and its flow among relevant stakeholders are analyzed. The following analysis shows data flow & volume between stakeholders for digital twin in each use case.

3.4.1. Area Management Security

One example of AMS is an airport use case which uses several digital twins. As the results, data structure and flow of AMS in airport are shown in below.

Stakeholders and data structure of an airport digital twin are shown below.

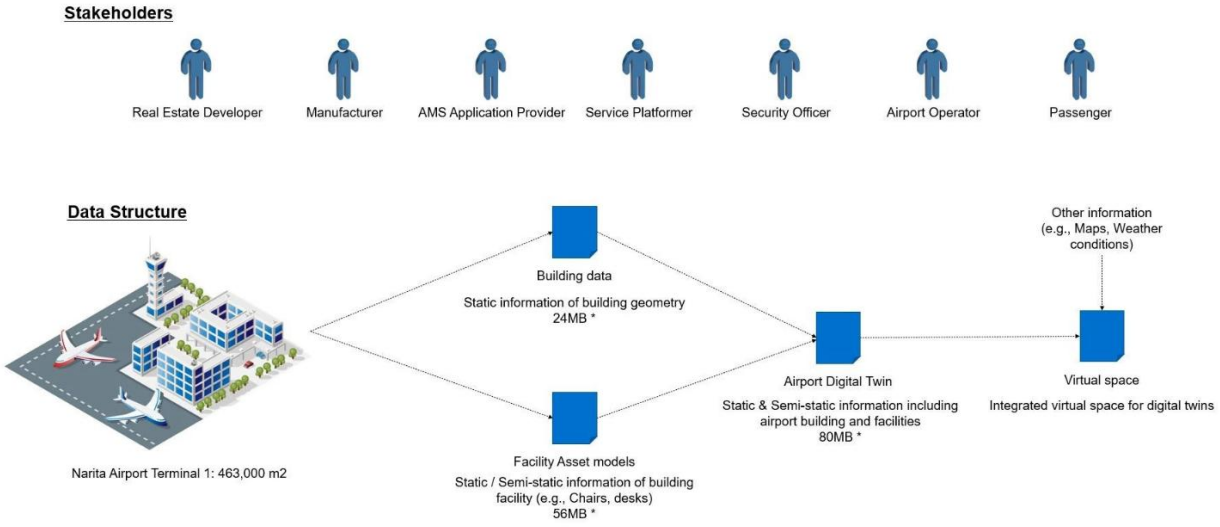

Figure 3.4-1: Stakeholders and data structure of airport digital twin

**Estimation of Live4D map volume & velocity: [IOWN IDH]*

Data flow of an airport digital twin based on lifecycle (create, update, and consume) is shown below.

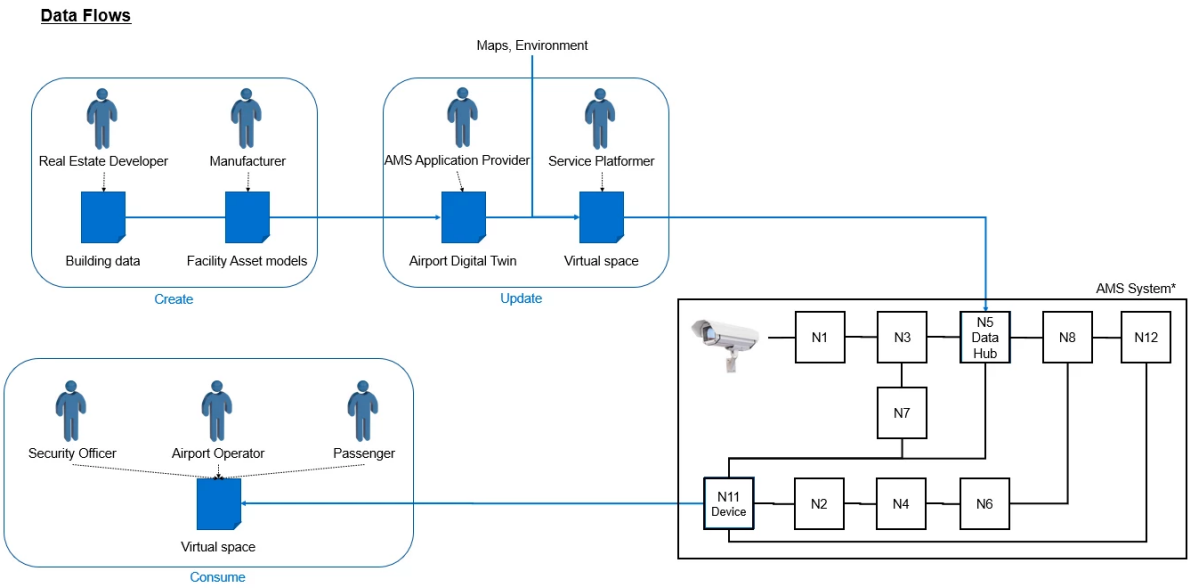

Figure 3.4-2: Data flows of airport digital twin

**According to Reference Implementation Model (RIM) for the Area Management Security Use Case, Figure 3.1-1: A Data Pipeline Diagram for AM Security UC: [IOWN RIM AM]*

As shown above, an airport digital twin is created and updated by a real estate developer, a manufacturer, and an AMS application provider as a building block of digital twins. Then, it will be consumed by application users such as a Security Officer and an Airport Operator. As this digital twin is relatively simple and static, the digital twin data is stored in Data Hub node as pre-condition and won't be updated frequently.

Digital Twin Example: Suspicious Objects (Dynamic object digital twin)

Stakeholders and data structure of a digital twin of a suspicious object are shown below.

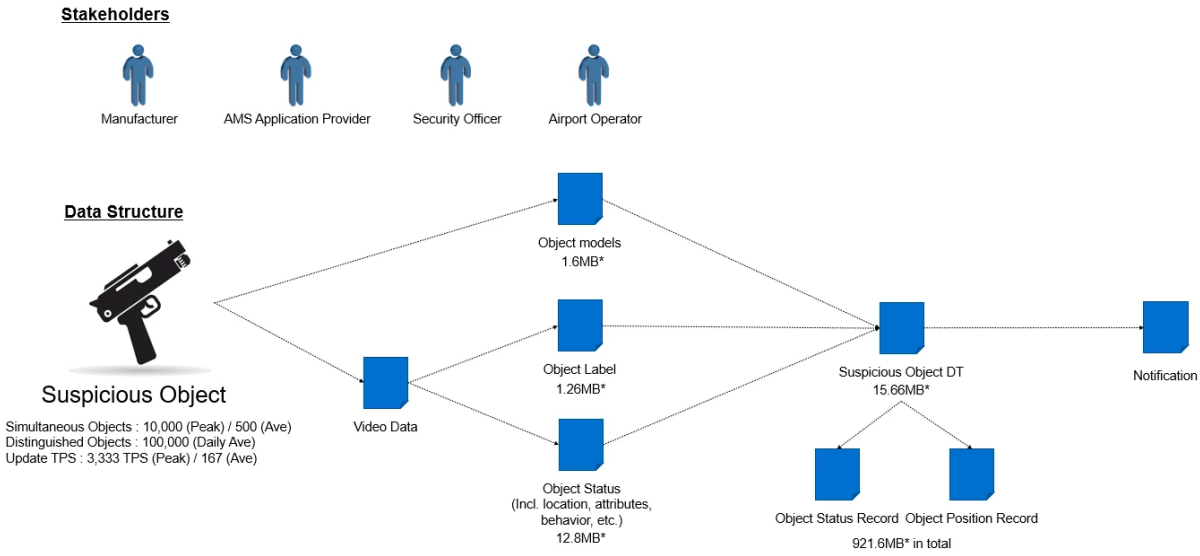

Figure 3.4-3: Stakeholders and data structure of suspicious object digital twin

**Estimation of Live4D map volume & velocity: [IOWN IDH]*

Data flow of a digital twin of a suspicious object based on lifecycle (create, update, and consume) is shown below.

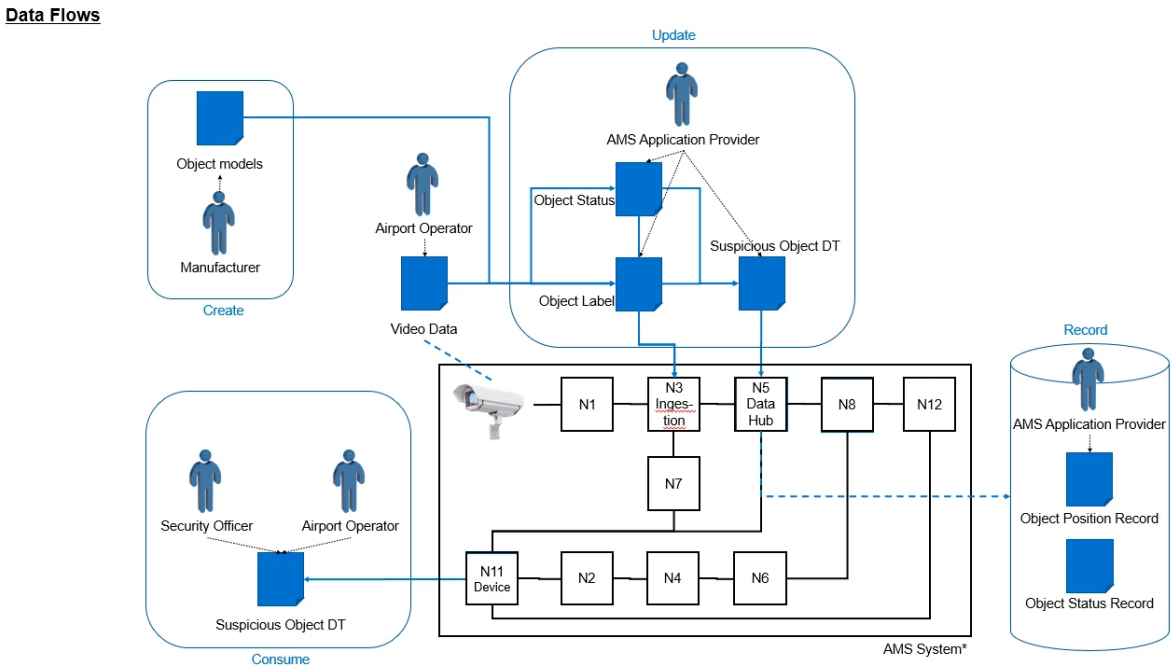

Figure 3.4-4: Data flows of suspicious object digital twin

**According to Reference Implementation Model (RIM) for the Area Management Security Use Case, Figure 3.1-1: A Data Pipeline Diagram for AM Security UC: [IOWN RIM AM]*

As shown above, the digital twin of a suspicious object is created and updated by a manufacturer, an airport operator, and an AMS application provider as a specific digital twin to indicate an individual object. In addition, an AMS application provider stores time-series data and logs in their data base. Then, the digital twin is consumed by application users such as a Security Officer and an Airport Operator. As this digital twin is dynamic, the digital twin data is updated continuously and stored in Data Hub node frequently.

3.4.2. Green Twin

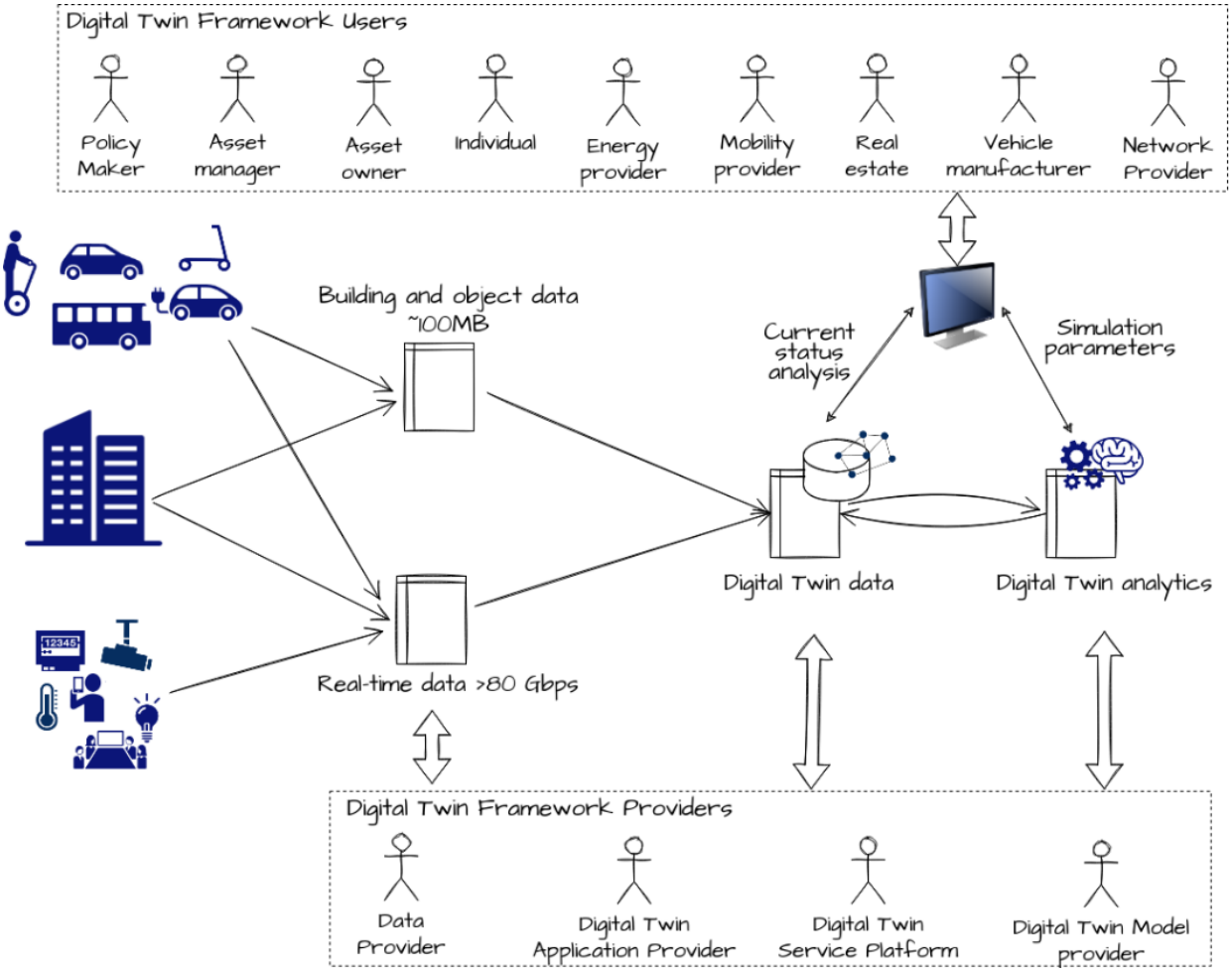

Figure 3.4-5: Structure and data flow of Green Twin

In the figure above, it is shown how the digital twin for the Green Twin (and in particular for the building scenario) is built. Two types of data are considered in this figure: Static/semi-static data and real-time data.

For this analysis we take as a reference a university building of the Campus of University of Murcia (Spain) (https://www.um.es/web/universidad/mapas/medicina) composed of 6 levels (2 of which are underground), 500 rooms, and 40 hallways.

- Static or semi-static data might reach a total amount of circa 100 MB stored at one time.
  - 3D models of the building,
  - HVAC configuration and datasheets.

  - Electric appliances datasheets
- The real-time data is generated every second by a multitude of sensors deployed in the building reaching circa 80 Gbps:
  - Environments sensors (e.g., video cameras, occupancy, motion sensor, light sensor, temperature, sensor, power sensor into electricity sockets),
  - Smart meters (power meters, water meters, wastewater meters, solar panel monitoring),
  - Outdoor environment (parking sensors, weather, noise sensor)
  - Network monitoring (5G, optical, Wi-Fi)
  - Smart appliances (monitors, coffee machines, vending machines, gates/doors)
  - Wearables on individuals (inertial sensors, vital sign sensors)

The data is, then, linked together and stored to have a data representation of the building digital twin. On top of this data, analytics processes give life to the building in the digital world. The behavior of the building is replicated in all its aspects and complexity. These analytics infer current situations, predict future situations, and allow simulation for hypothetical conditions. The analytics results augment the Digital Twin data and are used by another analytics process in a continuous process.

Stakeholders can interact with the green twin through an interface that allows them to visualize the up-to-date status of the building or simulate hypothetical scenarios. For example:

- Policymakers might inspect actual power consumption and simulate expected power consumption if there is a change to office hours range or working hours constraints.
- Asset managers or asset owners might inspect power (or water) consumption and simulate the consequences of changing HVAC configurations, upgrading HVAC systems, changing room configurations (disposition or destination), changing electricity schedule to use self-generated solar energy, etc.
- Individuals might interact with green twin to have an optimized schedule of their activity to reduce energy impact (e.g., use rooms already heated/cooled, use already lighted because of other people, optimize working schedule to reduce parking time)
- Energy providers might inspect the energy consumption of the green twin of the building to optimize the power grid (e.g., reduce capabilities for that specific area if the solar panels are already provisioning a good portion of the needed energy for the building)
- Real estate might inspect the energy behavior of the building to simulate the energy efficiency of similar buildings.
- Mobility providers might simulate the behavior in terms of performances (e.g., customer satisfaction, optimal usage of the fleet) and energy consumption of (public) transportation in case of a change of schedule, change of traffic circulation, or change of policies (e.g., maintenance schedule or fleet size).
- Vehicle manufacturers might infer the requirements of customers in a smart city environment and use maintenance prediction analytics to better organize customer support and improve logistics and customer satisfaction.

Further, other stakeholders provide the digital twin framework and its instance. For example:

- Data providers are necessary to populate the data digital twin and enable the analytics
- Digital Twin application providers are integrating and maintaining different pieces together for a specific application generating views and features
- Digital Twin service platform providers implement and maintain the computing nodes, the network, and the software for the data storage and the analytics execution environment
- Digital Twin model providers implement the behavioral model and analytics of the digital twin.

3.4.3. Area Management Disaster Notification

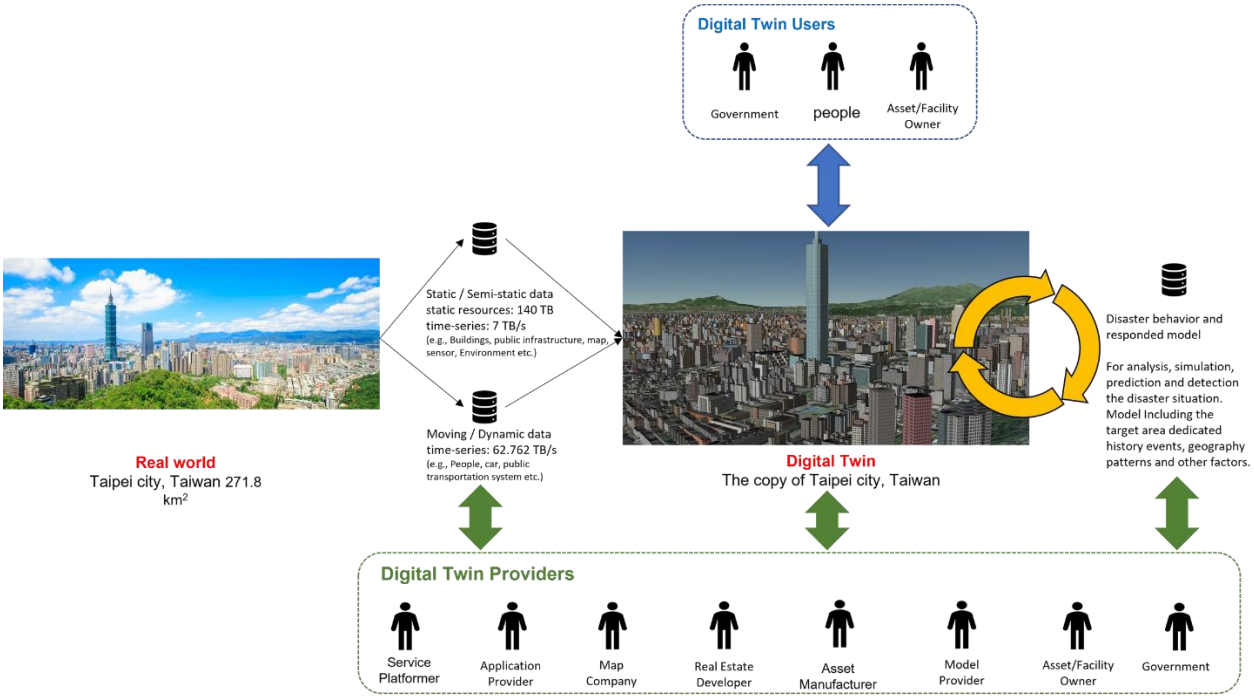

Figure 3.4-6: Digital Twin of city for disaster notification

In the figure above, it descripts the data structure and data flow of digital twin for the disaster notification use case which contains types of data, data model as well as relationship of stakeholders for providing and using this digital twin.

Data types – Necessary data for building the digital twin world

- Static and semi-static data
  - Static resources need about 140TB
    - 3D building models (BIM) with LOD 400 above.
    - The sensing facility around 500Km of Target area
    - 3D area map
  - Time series reach about 7TB/s
    - Sensors inside and around the outside of the buildings
    - Dedicated sensors for disaster detection
    - Environment situation
- Moving / Dynamic Objects
  - Time series reach about 62.762TB/s
    - People's position, moving path
    - The position, moving path and status of vehicles, transportation system and anything that takes the people
    - Emergency unit's position, status, and vital sign, etc.

Data model – The brain of digital twin

- Disaster behavior and responded model

- For analysis, simulation, prediction, and detection the disaster situation.
- Respond to disaster situations, such as stopping trains, indicating escape routes, etc.
- Model Includes the target area dedicated history events, geography patterns and other factors.

Stakeholder – Who creating, updating and using the digital twin

- Digital Twin Providers
  - Service platformer provides the network infrastructure, data gathering and storage service, computing, etc.
  - Application provider provides the disaster notification application service for users.
  - Map company provides the related of geography data (e.g., terrain, map layer, satellite image, etc.) in the target area.
  - Real Estate Developer provides the building geometry model, including interior and exterior assets, material, etc.
  - Asset Manufacturer who provides the asset's geometry model and information. (e.g., car, train, machine etc.)
  - Model Provider providers implement the behavioral model and analytics of the digital twin.
  - Asset/Facility Owner provides asset/facility real time operation information.
  - Government provides necessary confidential and private information of nation for disaster notification application. (e.g., military/emergency unit deployment)
- Digital Twin Users
  - Government uses digital twin to simulate the situation when disaster strikes, to improve potential problems such as building strength. Also, government uses digital twin to deploy, make decision and get latest situation when the disaster happened.
  - People uses digital twin to get the alert when disaster coming and get escape route advice continually until people reach a safe place.
  - Asset/Facility Owner uses digital twin to get the alert when disaster coming and get advice to perform the emergency mechanism such as slow down the train, close the nuclear reactor, etc.

4. Requirements

4.1. Definition of digital twin composition

Through use case analysis, it is clarified that there exists several characteristics of digital twin for use cases in IOWN GF. Although many types of digital twins are used in different scenarios, the characteristics can be categorized into three types as shown in Figure 4.1-1.

The first type of characteristic is the pace of change of digital twins. According to the analysis for patterns of digital twins, static, semi-static, and dynamic (could be separated into semi-dynamic and dynamic) digital twins are used in use cases. A static digital twin means a digital twin that has low frequency of changes, and the data consisting of the digital twin is relatively static, while a dynamic digital twin (including process digital twin) changes dynamically and has frequent updates on its data such as position, status, and behavior that produce a lot of historical data (e.g., previous states/information) as well. This type is the basic categories of digital twins and affects other features described below. In addition, this category could change based on duration we monitor a digital twin. So, duration of use case could be an important factor to decide the category.

The second characteristic is the scale or size of digital twins. There might exist certain relation between the pace of change and scale of a static digital twin such as a large facility, a building, and environment of target area in general. On the other hand, a dynamic digital twin is a relatively small size object such as a robot, a vehicle, and a human. These small size digital twins can be used in large facilities or buildings as dynamic objects in the IOWN GF use cases.

The third characteristic is data composition that is a ratio of fixed data and time-series data (e.g., percentage, volume of data). Considering natures of static digital twins and dynamic digital twins, the static one consists of mostly fixed data that is defined as a model or design data and is not modified frequently, while the dynamic one contains a lot of time series data to represent its history of changes. The fixed data includes map, building design, facility asset, geometry, material, and metadata that does not change over time. The time series data typically includes position/trace data, behavior, status, energy consumption, sensing/camera data and environmental conditions according to the analysis in the previous sections.

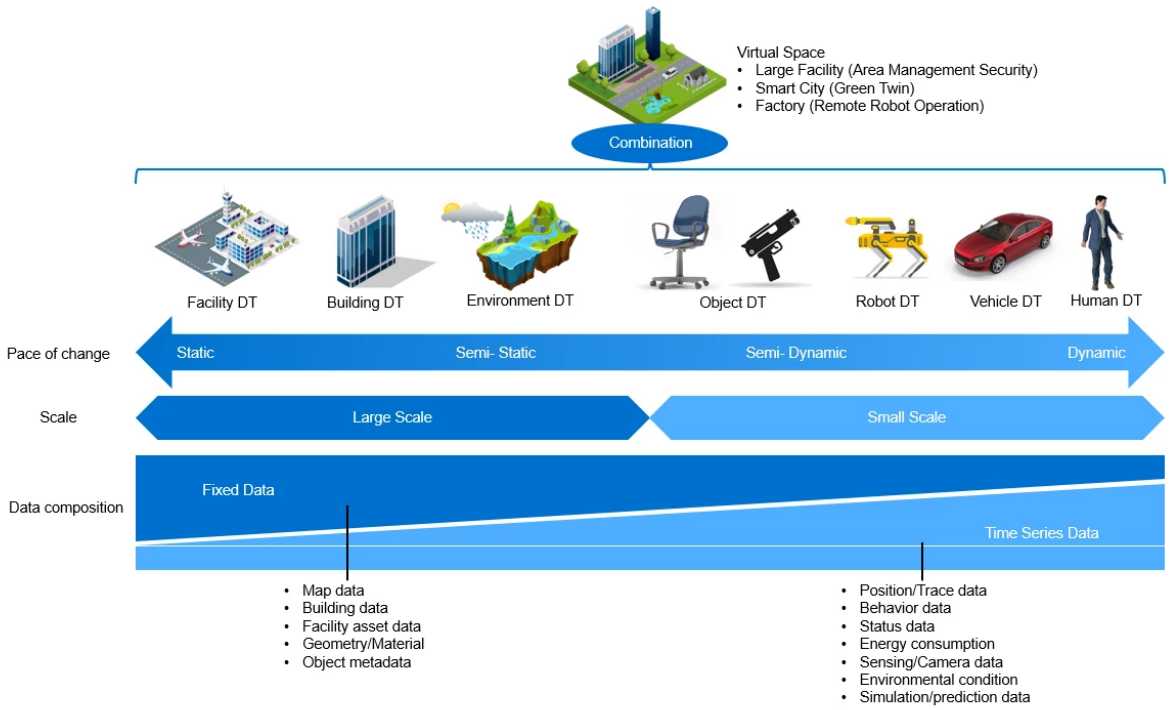

Figure 4.1-1: Composition of digital twins

These characteristics represent typical types of digital twins used in IOWN GF use cases. In addition, most use cases require a combination of several digital twins having different features like system of systems. For examples, the Area Management Security requires static digital twins representing a large target area that includes several digital twins of facilities and buildings with dynamic digital twins of a suspicious object and pedestrians. The Green Twin also requires static digital twins of buildings with dynamic digital twins of energy consumption of vehicles and individuals together.

An important point is that necessary data and requirements for compute and network infrastructure are affected by composition of digital twins used in a use case. If the use case needs mostly static and big scale digital twins, their data is mostly fixed data and is not frequently modified. Even if a use case requires large capacity of data storage, there is no strong requirement for uploading new data through networks and writing new data on databases as the data is relatively stable. On the other hand, if a use case contains a lot of dynamic digital twins, frequently data transmission through telecommunication infrastructure and data storages are required to manage many transactions of data to be updated dynamic digital twins.

According to analysis of digital twin composition, the IOWN GF needs to address two points as follows;

- The first requirement is that the IOWN GF infrastructure, including APN and DCI, is required to be flexible to accommodate various types of use cases that have critical differences of static and dynamic composition of digital twins.
- The second requirement is that the DTF needs to find a way to define static and dynamic composition of digital twins based on use case scenarios and preconditions to identify concrete requirements for the IOWN GF infrastructures.

4.2. Mechanism of cross-domain data flows

From the results of analysis on the Area Management Security and the Green Twin use case, the cross-domain data flows which mean digital twin data is passed between many stakeholders from different domains (e.g., companies, industries) are recognized. In the AMS case, a digital twin of building is created and updated by a Real Estate Developer and Facility Operator who are in real estate domain, then the digital twin is passed to a Security Company and a Person (a user) of the building who are in Service Provider and User domain. In the case of the Green Twin, a digital twin of vehicle is created by Manufacturer, then the digital twin is shared to an Asset Owner and a Policy Maker, who are also in different domain, for Update and Consume.

This cross-domain data flows are observed in the Open Area use case *2 in general. The Open Area use case includes smart community, smart city, and smart environment. Due to multiple stakeholders in this use case, digital twin data needs to be shared among many stakeholders, and this situation creates a cross-domain data flows between stakeholders from different domains during the lifecycle of a digital twin.

Considering the cross-domain data flows which commonly happens in the Open Area use cases such as smart cities, area managements, and municipal services, as the general use cases of digital twins, the way of providing a digital twin platform which handling the cross-domain data flows in terms of data access right, privacy management, and efficiency of data exchange are important requirements to be addressed in the IOWN GF.

**2 see "Definitions" section for details.*

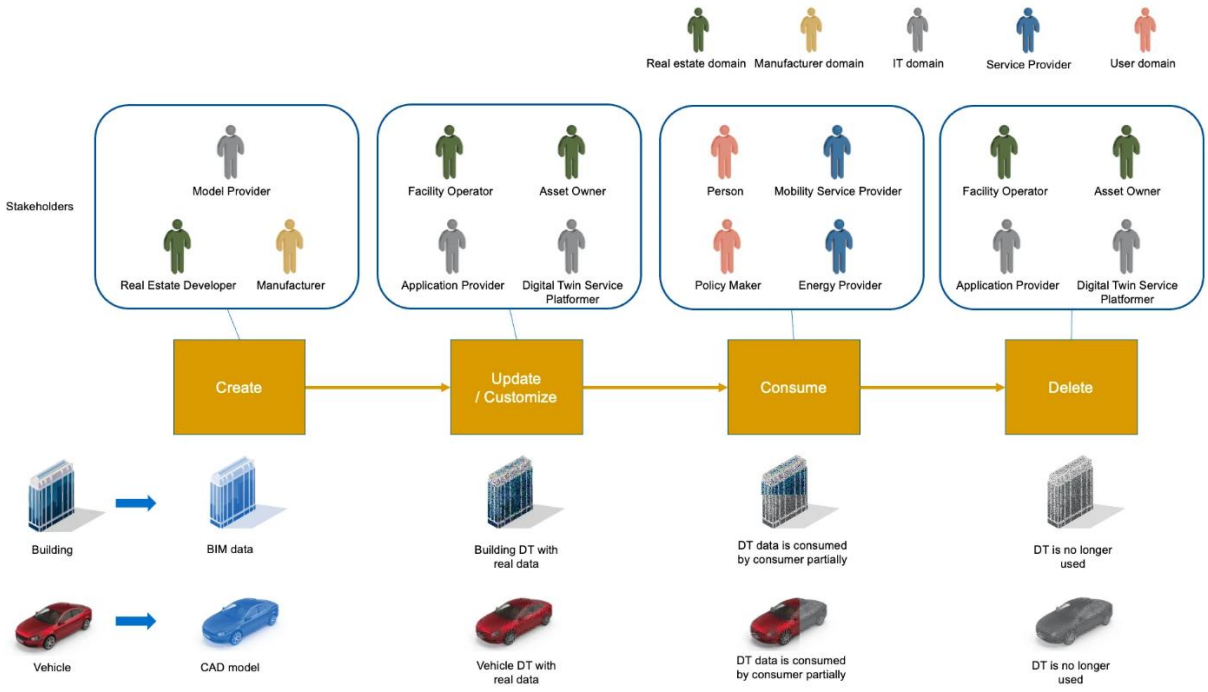

Figure 4.2-1: Cross-domain data flows in Open Area use-case (AMS and Green Twin)

On the other hand, different patterns of digital twin use cases are identified as the Closed Area use case *3 such as the Remote Robot Operation and the Human digital twin. In case of the Remote Robot Operation, entire system is designed for a Factory Operator to control remote robots, and limited number of stakeholders who are in the same supply chain with relevant domains share digital twin data. This Closed Area use cases are also typical patterns of digital twins and have limited requirement for cross-domain data flows.

It is required to identify a pattern of a use case (the Open Area use case or the Closed Area use case) to create appropriate system architecture, taking into account an impact on system requirements when the cross-domain data flows exist.

Also, interoperability of digital twins is required in both types of use cases. For example, a digital twin of electric vehicle created by a manufacturer needs to collaborate with an EV station digital twin created by an energy provider, while a robot digital twin created by a robot manufacturer needs to be used in a digital twin of a factory. Although interoperability between digital twins can be supported by creating common models, interoperable interfaces, or creating a mechanism bridging different data models, the approach needs to be clarified and agreed by relevant stakeholders.

**3 see "Definitions" section for details.*

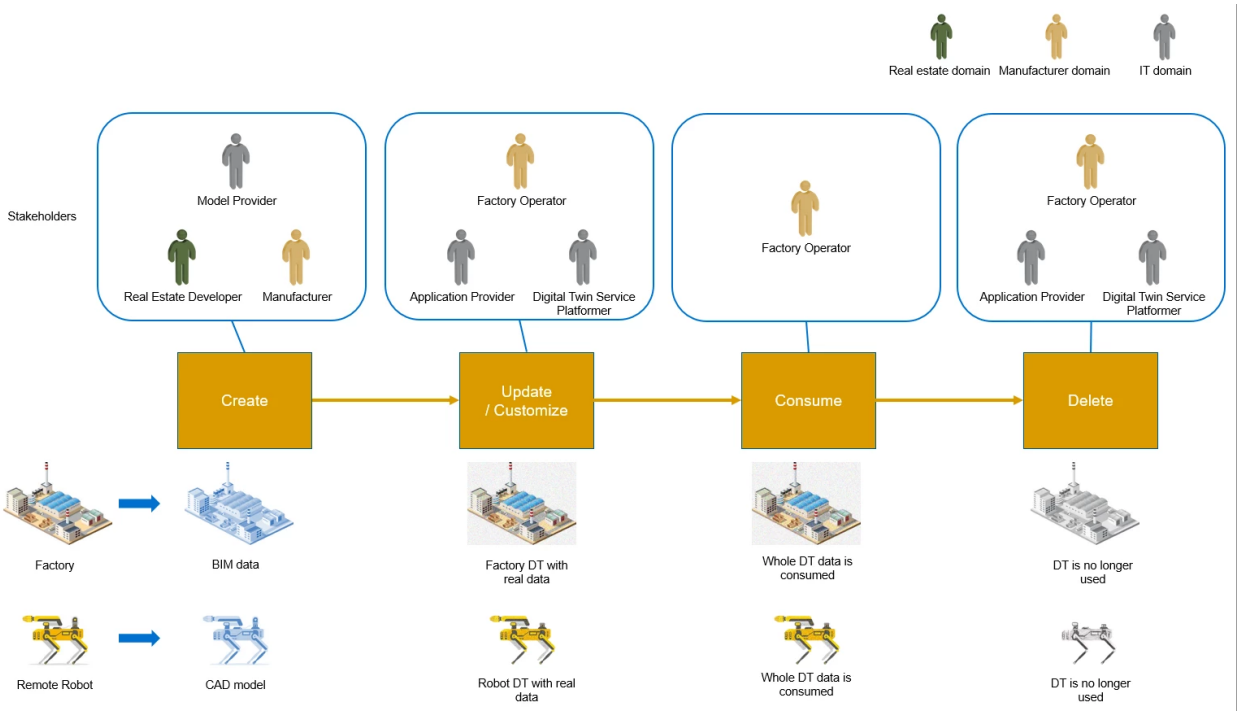

Figure 4.2-2: Cross-domain data flows in Closed Area use-case (e.g., Remote Robot Operation, Human DT)

In addition, there is another issue of cross-domain data flows. As shown in Figure 4.2-3, a vehicle digital twin can include data of exterior structure, interior structure, and attributes. In general, a digital twin consists of several types of data. The exterior structure can be shared with many stakeholders, while the interior structure can be shared with very limited number of stakeholders due to information confidentiality of a manufacturer. Attribute data including tax and insurance information is very sensitive to be shared as they include private data.

A digital twin may contain different types of data in terms of confidentiality and privacy. Thus, it is required to develop a mechanism to manage access rights and privacy for a different part of a digital twin. In addition, it is required to find an effective mechanism to share necessary data included in a digital twin that contains different levels of confidentiality. This is also important issue to be addressed to create an interoperable digital twin among various types of stakeholders.

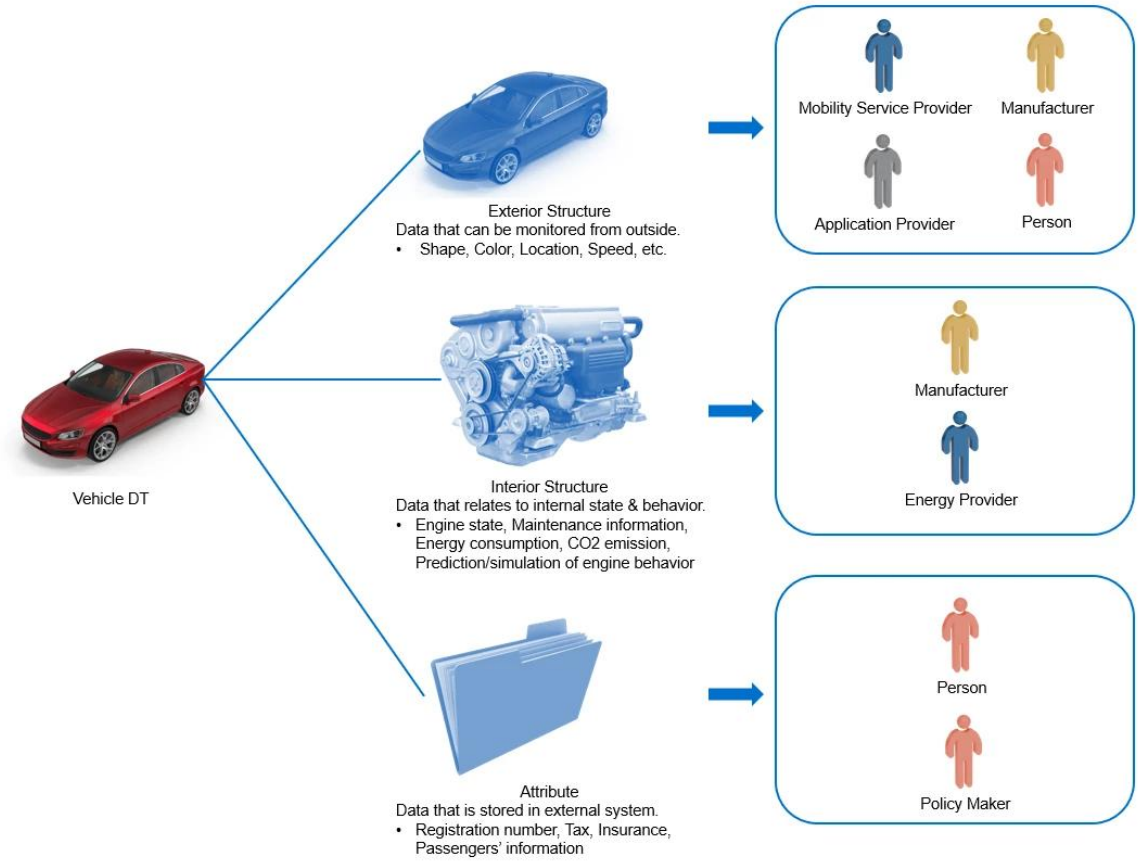

Figure 4.2-3: Vehicle digital twin composition

According to analysis of cross-domain data flows, the IOWN GF needs to address following points as follows;

- The first requirement is that the IOWN GF infrastructure needs to provide a platform to manage data flows of digital twins between different stakeholders. This requires access right management between stakeholders in terms of confidentiality, privacy, and efficiency of data exchange.
- The second requirement is IOWN GF needs to provide a mechanism to manage confidentiality and privacy of different parts of one digital twin. This could require definition for structure of a digital twin with explicit indication of confidentiality and could require collaboration with other SDOs.
- The third requirement is that the IOWN GF needs to identify an interoperability mechanism to interact different types of digital twins. This could require collaboration with other SDOs.
- The fourth requirement is the IOWN GF needs to provide a data usage control system to control how the data is used after it is accessed by a data consumer.

4.3. Volume of data

Data volume for relevant use cases such as the Area Management Security, Smart Factory and the Green Twin are estimated. These estimated volumes are the necessary data volumes for the first time creation of a digital twin.

Regarding the Area Management Security, the volume of digital twin data as well as input data that is processed to create a part of digital twin is analyzed. According to the estimate in the following page, the most of severe requirements should be the Surveillance video data (up to 58PB per monitored area). This video data is processed in a computing

node to produce a label and status information of target objects. The processed data are converged with the corresponding digital twin, so the digital twin data are much smaller than the input data.

Regarding the Remote Robot Operation, necessary data volume based on its use case scenario is estimated. The estimated data volume could change in future as the Reference Implementation Model is still under development. According to the estimation of smart factories, the most severe requirements come from input data such as sensor data (up to 526PB for 10 years) and capturing images (up to 303PB for 10 years). Since digital twins are created by processing these input data, data volume of digital twin itself is much smaller.

Regarding the Green Twin, the necessary volume based on the different twin types: Smart Building Twin, Person Twin, and Vehicle Twin is estimated. The estimated data volume could change in the future based on the use case development. For this analysis we take as reference a university building of the Campus of Murcia [UniMurcia Facultad Medicina] composed of 6 levels (2 of which are underground), 500 rooms, and 40 hallways. The data requirements for the vehicle refer to the data requirements defined in the relevant IOWN Use Case Document Release 1. This table includes the calculations based on the listed sensors in this vehicle use case. Similarly, the sensors that are expected for Smart Building and Person Twin are listed. For the Person Twin, an urban environment with sensors such as cameras is considered, where the same data velocity may serve multiple Persons. Lastly, as in the previous cases, the severe requirements come from long-term measurements such as 1 year in an urban environment (e.g., city square, airport, or train station) and 20 years for a smart building.

5. Gap Analysis

During the Use case analysis, six requirements are identified.

- DTF-Req-1: The IOWN GF infrastructure, including APN and DCI, is required to be flexible to accommodate various types of use cases that have critical differences of static and dynamic composition of digital twins.
- DTF-Req-2: The DTF needs to find a way to define static and dynamic composition of digital twins based on use case scenarios and preconditions to identify concrete requirements for the IOWN GF infrastructures.
- DTF-Req-3: The IOWN GF infrastructure needs to provide a platform to manage data flows of digital twins between different stakeholders. This requires access right management between stakeholders in terms of confidentiality, privacy, and efficiency of data exchange.
- DTF-Req-4: The IOWN GF needs to provide a mechanism to manage confidentiality and privacy of different parts of one digital twin. This could be a guideline for structure of a digital twin with explicit indication of confidentiality and could need to collaborate with other SDOs.
- DTF-Req-5: The IOWN GF needs to identify an interoperability mechanism to interact different types of digital twins. This could require collaboration with other SDOs.
- DTF-Req-6: The IOWN GF needs to provide a data usage control system to control how the data is used after it is accessed by a data consumer.

Among them, four requirements (DTF-Req-3 to DTF-Req-6) are particularly related to functions for data exchange and data processing. Therefore, a gap analysis in terms of digital twin framework is conducted focusing on these four requirements.

For gap analysis, these requirements are compared with existing technologies and related features. DTF-Req-3&4 focus data flow management with access right policy management between stakeholders. As Gaia-X has a similar concept, Gaia-X is chosen for this gap analysis. On the other hand, DTF-Req-5&6 are compared to Fiware for their gap analysis since interoperability of digital twin has been discussed in the Fiware especially in terms of information models. The detailed analysis result is shown below.

5.1. DTF-Req-3&4

DTF-Req-3&4 indicate necessity of access right policy management for accessing to data/attribute included in a digital twin in terms of confidentiality, privacy, and efficiency. The gaps between these requirements and the existing technologies are analyzed in this section, to find the key issue to fulfil these requirements.

5.1.1. Target Technology

The target technology of this gap analysis is specified in the Gaia-X as data exchange functions.

5.1.1.1. Reason

Gaia-X will provide holistic mechanism for data sharing, and many European consortiums are launching their data spaces based on the Gaia-X framework. Therefore, Gaia-X is selected as a target of this gap analysis. Gaia-X provides a platform for data exchange among many types of stakeholders by integrating relevant technologies from certification to access control. As this technology should provide ideal environment for DTF-Req-3&4 requirements, detailed gap analysis between existing Gaia-X functionalities and DTF-Req-3&4 requirements needs to be evaluated.

5.1.1.2. Feature of the target technology

Gaia-X is an association that addresses the multi-stakeholder data sharing. Gaia-X has an initiative to support data sharing via infrastructure that ensures data protection, transparency, reliability, interoperability, and ultimately data

sovereignty. It also creates an ecosystem of data spaces and starts specifying the data exchange solution such as IDS technologies using Eclipse Data Connector, which is an open source framework used to create connectors.

Figure 5.1-1 shows the basic mechanism of Gaia-X. As shown in the figure, Gaia-X provides the Federated Catalogue where users register metadata of their digital assets and relevant APIs to manage accesses from data consumers to the digital assets.

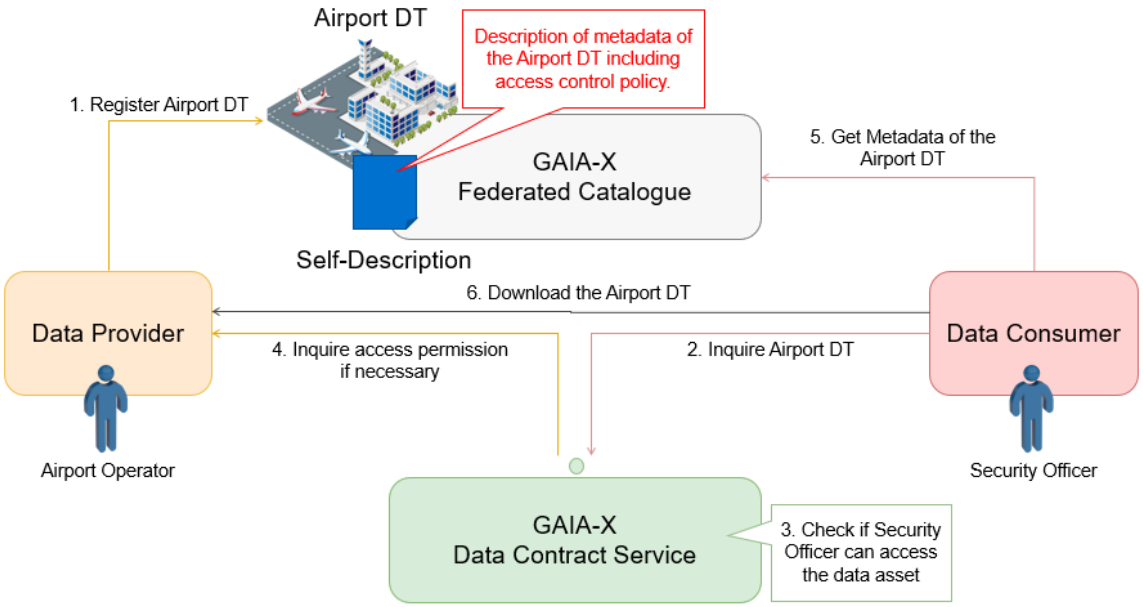

Figure 5.1-1: Basic mechanism of Gaia-X

However, specifications of Gaia-X are defined gradually, and the data exchange part, which is the most relevant feature for data sharing, has not had the official initial release yet as of 2022 September. Therefore, this gap analysis is based on limited perspective obtained from existing concept of Gaia-X. Also, the result and the conclusion of the gap analysis are possibly needed to be updated in the future once an initial specification of the Gaia-X data exchange has been released.

5.1.1.3. Points of gap analysis

Based on the requirements (DTF-Req-3&4), the following points are analyzed.

- Access right policy management for multiple stakeholders (corresponding to DTF-Req-3's confidentiality and privacy between stakeholders)
- Access right policy management for multiple attributes in single digital twin (corresponding to DTF-Req-4's confidentiality and privacy of different parts of one digital twin)
- Complexity of assignment for access right policy (corresponding to DTF-Req-3's efficiency of data exchange)

5.1.2. Gap Analysis

At first, typical pattern of access right policy is studied based on the use case analysis. This will be used to evaluate if Gaia-X provides sufficient functionalities.

Typical pattern of access right policy

According to the use case analysis, four typical patterns of access right policies are identified. In the use cases such as the Area Management Security and the Green Twin, there are many stakeholders from different domains/companies, and they exchange various types of digital twin data among them (Pattern 1, 2).

Due to digital twin lifecycle that includes Create, Update, Consume, and Delete phases processed by different stakeholders, the digital twin data is cascaded from one stakeholder to others. This situation is also identified as the inherited access right policy (Pattern 3).

In addition, the use cases include complex digital twins such as a building and a vehicle that contain multiple attributes with different access right policies. This is also typical pattern of digital twin use case scenario (Pattern 4).

- Pattern 1: Simply share data with restricted consumer

Table 5.1-1: Examples of Pattern 1

Provider	Data	Policy	Consumer	Action & Result
Airport Operator	Security Camera data *1 ● Attribute1: Video stream	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Application Provider, Security Officer	Application Provider	Access Security Camera data (Attribute1) : OK
Airport Operator	Security Camera data *1 ● Attribute1: Video stream	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Application Provider, Security Officer	Passenger	Access Security Camera data (Attribute1) : NG

**1 See Figure 3.4-2.*

- Pattern 2: Simply share data including multiple attributes with restricted consumer

Table 5.1-2: Examples of Pattern 2

Provider	Data	Policy	Consumer	Action & Result
Airport Operator	Airport DT *1 ● Attribute1: public area data ● Attribute2: private area data	Policy1: Permitted Consumer can access all attributes ● Permitted Consumer: Security Officer	Security Officer	Access Airport DT (Attribute1 & 2) : OK
Airport Operator	Airport DT *1 ● Attribute1: public area data ● Attribute2: private area data	Policy1: Permitted Consumer can access all attributes ● Permitted Consumer: Security Officer	Passenger	Access Airport DT (Attribute1) : NG
Manufacturer	Object Model (Suspicious Object) *2 ● Attribute1: Appearance ● Attribute2: Interior structure	Policy1: Permitted Consumer can access Attributes 1 & 2 ● Permitted Consumer: Application Provider Policy2: Permitted Consumer can access Attributes 1 ● Permitted Consumer: Security Officer	Application Provider	Access Object Model (Attribute1 & 2) : OK

Manufacturer	Object Model (Suspicious Object) *2 ● Attribute1: Appearance ● Attribute2: Interior structure	Policy1: Permitted Consumer can access Attributes 1 & 2 ● Permitted Consumer: Application Provider Policy2: Permitted Consumer can access Attributes 1 ● Permitted Consumer: Security Officer	Security Officer	Access Object Model (Attribute1 & 2) : NG

**1 See Figure 3.4-2. *2 See Figure 3.4-3.*

- Pattern 3: Share data including inherited attributes with restricted consumer

Table 5.1-3: Examples of Pattern 3

Provider	Data	Policy	Consumer	Action & Result
Application Provider	Suspicious Object DT *1 ● Attribute1: Appearance (inherited from Object Model of Manufacturer in Table 5.1-2) ● Attribute2: Interior structure (inherited from Object Model of Manufacturer in Table 5.1-2) ● Attribute3: Location, Behavior (newly added attribute by Application Provider)	Policy1: Permitted Consumer can access all attributes ● Permitted Consumer: Security Officer, Airport Operator	Security Officer	Access Suspicious Object DT (Attribute 1&3) : OK
Application Provider	Suspicious Object DT *1 ● Attribute1: Appearance (inherited from Object Model of Manufacturer in Table 5.1-2) ● Attribute2: Interior structure (inherited from Object Model of Manufacturer in Table 5.1-2) ● Attribute3: Location, Behavior (newly added attribute by Application Provider)	Policy1: Permitted Consumer can access all attributes ● Permitted Consumer: Security Officer, Airport Operator	Security Officer	Access Suspicious Object DT (Attribute 2) : OK Problem: The Attributes 2 is originally prohibited to be accessed by Security Officer in Table 5.1-2 but the policy not inherited automatically between stakeholders (in this case, between the Manufacturer and the Application Provider).

**1 See Figure 3.4-3.*

- Pattern 4: Share data including multiple attributes containing different policies

Table 5.1-4: Examples of Pattern 4

Provider	Data	Policy	Consumer	Action & Result
Asset Owner	Building DT *1 ● Attribute1: Exterior structure ● Attribute2: Interior structure ● Attribute3: Other information	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Map Company Policy2: Permitted Consumer can access all attributes ● Permitted Consumer: Facility Operator	Facility Operator	Access Building DT (Attribute2) (to manage a defect in the building) : OK
Asset Owner	Building DT *1 ● Attribute1: Exterior structure ● Attribute2: Interior structure ● Attribute3: Other information	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Map Company Policy2: Permitted Consumer can access all attributes ● Permitted Consumer: Facility Operator	Map Company	Access Building DT (Attribute1) (to create a map) : OK
Asset Owner	Building DT *1 ● Attribute1: Exterior structure ● Attribute2: Interior structure ● Attribute3: Other information	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Map Company Policy2: Permitted Consumer can access all attributes ● Permitted Consumer: Facility Operator	Map Company	Access Building DT (Attribute2) (to create a map) : NG
Car Manufacturer	Vehicle DT *1 ● Attribute1: Exterior structure ● Attribute2: Interior structure ● Attribute3: Other information	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Passenger Policy2: Permitted Consumer can access Attribute1 & 3 ● Permitted Consumer: Energy Provider Policy3: Permitted Consumer can access all attributes ● Permitted Consumer: Manufacturer	Passenger	Access Vehicle DT (Attribute1) : OK

Car Manufacturer	Vehicle DT *1 ● Attribute1: Exterior structure ● Attribute2: Interior structure ● Attribute3: Other information	Policy1: Permitted Consumer can access Attribute1 ● Permitted Consumer: Passenger Policy2: Permitted Consumer can access Attribute1 & 3 ● Permitted Consumer: Energy Provider Policy3: Permitted Consumer can access all attributes ● Permitted Consumer: Manufacturer	Energy Provider	Access Vehicle DT (Attribute3: energy consumption) : OK

**1 See Figure 4.1-1.*

5.1.2.1. Evaluation: Access right policy management for multiple stakeholders

Regarding Pattern 1 & 2, these are basic access control patterns in terms of confidentiality and privacy. Like most of other access control technologies, Gaia-X also provides fundamental function to support access right policy required by these patterns according to the existing specifications. So there is no gap between requirements and Gaia-X.

Due to digital twin lifecycle that includes Create, Update, Consume, and Delete phases processed by different stakeholders, the digital twin data is cascaded from one stakeholder to others. This situation is also identified as the inherited access right policy (Pattern 3). Without an appropriate policy control mechanism, inconsistent policies that conflict with the entire digital twin may be defined on the data passed from these former stakeholders. To solve this issue, a mechanism to manage inherited access right policies should be needed. For example, when a stakeholder adds/modifies a new attribute, new policy that will be inherited from stakeholders in upstream is also created and maintained. As current Gaia-X specification does not have such function, this is a possible gap between the requirement and the existing technology.

Pattern 4 is discussed in the next section.

5.1.2.2. Evaluation: Access control for multiple attributes in single digital twin

In case that a digital twin includes multiple attributes with different access right policies (Pattern 4), access control needs to be done by each attribute with different policy. This requires complicated assignment of access right policy that manages a policy per an attribute of a digital twin, but access control mechanism is not so difficult to be implemented.

It is expected that Gaia-X also provides detailed policy assignment supporting access control for multiple attributes in single digital twin. In addition, this should be more a digital twin issue than a Gaia-X insufficiency. Therefore, there is no apparent gap from this point of view.

5.1.2.3. Evaluation: Complexity of assignment for access right policy

There is another point of view for the case that a digital twin includes multiple attributes. The point is assignment operation of access right policy. The assignment of access right policy for multiple attributes is a complicated task and requires huge effort, although the assignment for multiple attributes is possible as described in the above evaluation. In addition, it is difficult to consider the assignment of access right policy for a consumer who is not identified when the digital twin is created.

To solve this issue, a mechanism to simplify policy assignment such as Data Spaces and Role Based Access Control (RBAC) needs to be created. This is a possible gap between the requirement and the existing technology since GAIA-X has not clearly defined such mechanism in their specifications.

5.1.3. Identified Gaps

Following two gaps are identified through the gap analysis for DTF-Req-3&4 requirements.

- DTF-Gap-1: A mechanism to manage inherited access right policies that are created by previous stakeholders is not identified. As a digital twin contains many types of data added/modified by different stakeholders, accompanied access right policies need to be inherited from the original digital twin to the latest digital twin.
- DTF-Gap-2: Policy assignment operation needs to be simplified. As a digital twin consists of many types of data that are consumed by different stakeholders, assigning and maintaining precise access right policies require huge cost.

5.1.4. Conclusion

In this gap analysis, two gaps are identified. This gap analysis focuses on the access right policy management that is important to realize cross-industry exchange of digital twins. In the IOWN GF use cases that include the Area Management Security and the Green Twin, there are a lot of digital twins in their applications that come from many different industry domains. Therefore, the identified gaps relating to the implementation of the access right policy management are important to combine these digital twins with interoperable way, and should be resolved in the next step in the IOWN GF.

5.2. DTF-Req-5&6

The IOWN GF needs to identify an interoperability mechanism to interact different types of digital twins. Different types of digital twins might be handled by different stakeholders, thus, digital twin data and digital twin analytics need to be federated. This section identifies gaps needed to fulfill the interoperability requirement. Further, the IOWN GF needs to provide a data usage control system to control how the data is used after it is accessed by a data consumer.

5.2.1. Target Technology

The target technology of this gap analysis is FIWARE. FIWARE is a curated framework of components, all open sources, targeting the development of smart solutions.

5.2.1.1. Reason

Since its foundation, FIWARE has worked to simplify the development of solutions taking leverage of open standards and implementation-driven software platform. The core element of the information model of FIWARE is the *context* that is data with metadata that describes a situation or thing in the real world. In past two years, FIWARE has been focused on the Digital Twin aspect as one of the main use cases for further development, since Digital Twin is the natural evolution of the context-based information model. FIWARE solutions are meant to work in federated and distributed systems with the integration of heterogeneous and legacy systems. The core information manager of FIWARE is the Context Broker that is standardized by the Industry Specification Group (ISG) cross cutting Context Information Management (CIM) [ETSI GS CIM 009] from European Telecommunications Standards Institute (ETSI). The Context Broker is already one of the IOWN Data Hub service classes [IOWN IDH].

5.2.1.2. Feature of the target technology

This study focuses on FIWARE interoperability features between different types of digital twins and different digital twin stakeholders, such as Smart Data Models, Context Broker Architecture, Digital Twin analytics, Data usage control, and Data analytics orchestration.

NGSI-LD and Smart Data Models

The information model of FIWARE is based on the model provided by the ETSI ISG CIM standard, namely Next Generation Service Interface-Linked Data (NGSI-LD) information model [ETSI GS CIM 009]. The NGSI-LD information is based on graph and it is formed by three parts:

- Core metadata model
- Cross domain ontology
- Domain-specific ontology

The core meta model specifies the formal structure of the graph. It encompasses 4 classes:

- Entity. NGSI-LD information model is used for *context* and the data is represented bond to an object that is an entity (e.g., a room, a building, a machine).
- Relationship is a link between entities that define information such as *roomA is in buildingX*
- Property is a piece of information related to an entity
- Value is the value of the piece of information

Cross-domain ontology specifies information that are common among domains, such as *geoProperty*, *temporalProperty, unitCode*, *modified At, observedAt, createdAt, etc.*

The NGSI-LD information model does not specify any domain-specific ontology, hence giving freedom to the application developer to the best suited application data model. However, Smart Data Models [Smart Data Models] initiative aims at accelerating the development of the domain-specific ontologies. Smart Data Models initiative is supported by FIWARE, Open and Agile Smart Cities (OASC), tmForum, and India Urban Data Exchange (IUDX). Smart Data Models offers a very wide set of domain-specific data models compatible with NGSI-LD, for example in the domain of: Smart Cities, energy domain, environment domain, manufacturing domain, robotics domain, water Domain, aeronautics, agrifood domain, health, and logistics.

Context Broker Architecture

The context broker is the main component for the handling of data into a FIWARE based platform. Its architecture is also defined by the ETSI ISG CIM [ETSI GS CIM 009]. The architecture is versatile and it can be centralized, distributed, and federated.

Digital Twin analytics

There are already many FIWARE architecture to support early Digital Twin solutions. However, there is a current study to have a standardized model to integrate a Digital Twin system, information model and data processing tasks. This study is under the ETSI ISG CIM work.

Data analytics orchestration

In Digital Twin approach, data analytics processes run continuously on the data enhancing the digital twin with predictions (current, future or hypothetical). The orchestration of many analytics and the interaction between them is a challenge to be addressed. FIWARE is offering a system, namely FogFlow [FogFlow], to orchestrate such analytics keeping the configuration burden for the service developer to the minimum.

Data Usage Control

Data usage control encompasses both the access rights control and the how the data is used after the data had been accessed by a data consumer. FIWARE does not offer data usage control systems, however, some research studies and prototypes have been implemented starting from FIWARE information model and other FIWARE components. We check this feature from the point of view of interoperability between stakeholders.

5.2.1.3. Points of gap analysis

We analyze the mentioned FIWARE features in the light of the Digital Twin Framework use cases identified. The points we study are:

- The support of the information model for the identified IOWN Digital Twin use cases
- The support for a versatile distributed and federated data management system
- The support for handling data processing systems together with Digital Twin data
- The support for orchestration of data analytics in a cross-domain and distributed shared infrastructure.
- Configuration complexity to enforce data usage control and execute compliant data consumer services.
- Overhead affecting the execution of analytics for digital twin when enforcing data usage control.

5.2.2. Gap Analysis

5.2.2.1. NGSI-LD and Smart Data Models

The NGSI-LD information model addresses well the needs for modelling the digital twin data. Here we have a mapping of Digital Twin data to NGSI-LD data model:

- Digital Twin instance → NGSI-LD Entity
  - NGSI-LD and a Digital Twin instance are object-centric. All the NGSI-LD piece of information refers to an object that is modeled by an NGSI-LD entity. A Digital Twin is also object-centric, because a digital twin is a digital object that represents a physical object.
- Digital Twin attribute → NGSI-LD property
  - A property always refers to an object.
  - In the simplest case an object is a device/sensor
- Relationships between Digital Twin instances (also of different types) → NGSI-LD Relationship
  - Two Digital Twin instances, also if different types, are related to each other. For example, a digital twin instance might be part of another digital twin instance (e.g., a room into a building, a component into a machine). The NGSI-LD relationship element is perfectly suited to model these relationships.
- Relationships between sensors and devices to a Digital Twin instance → NGSI-LD Relationship
  - Link between Digital Twin (e.g., building) and Sensor/Device can be specified by this special field.

We have, then, explored the Smart Data Models to identify if the current ones support well the IOWN GF use cases discussed in the use case analysis section of this report.

Table 5.2-1: Green Twin: Smart Data Models analysis

Data	Smart Data Models	Link	Comments
Camera	Camera	https://github.com/smart-data-models/dataModel.Device/tree/master/Camera	Comprises of external links to stream and/or snapshot
3D Building	<missing>	-	

Sensors	Device	https://smart-data-models.github.io/dataModel.Device/Device	Generic for Device. There are many other specific data models, e.g., air quality: https://github.com/smart-data-models/dataModel.OCF/tree/master/AirQuality
HVAC	<missing>	-	
Human presence	Presence	https://github.com/smart-data-models/dataModel.OCF/tree/master/Presence	
Events	Event LifeEvent	https://smart-data-models.github.io/dataModel.TourismDestinations/Event/examples/example.jsonld https://github.com/smart-data-models/dataModel.CPSV-AP/tree/master/LifeEvent	
Vehicle	FleetVehicle FleetVehicleStatus FleetVehicleOperation Vehicle VehicleModel VehicleFault	https://github.com/smart-data-models/dataModel.Transportation/tree/master/Vehicle https://github.com/smart-data-models/dataModel.Transportation/tree/master/FleetVehicle …etc.	
Vehicle Route	<missing>	-	
LiDAR	<missing>	-	
Human Mobility	CrowdFlowObserved	https://github.com/smart-data-models/dataModel.Transportation/tree/master/CrowdFlowObserved	
Wearable	<missing>	-	
Weather	WeatherAlert WeatherForecast WeatherObserved	https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherAlert https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherForecast https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherObserved	

Table 5.2-2: Area Management: Smart Data Models analysis

Data	Smart Data Models	Link	Comments

3D Building	<missing>	-	
Area Map	GIS Data <partial match>	https://smart-data-models.github.io/dataModel.RiskManagement/GISData/examples/example.jsonld	This might be not a perfect match
Weather	WeatherAlert WeatherForecast WeatherObserved	https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherAlert https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherForecast https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherObserved	
Events	Event LifeEvent	https://smart-data-models.github.io/dataModel.TourismDestinations/Event/examples/example.jsonld https://github.com/smart-data-models/dataModel.CPSV-AP/tree/master/LifeEvent	
Camera	Camera	https://github.com/smart-data-models/dataModel.Device/tree/master/Camera	Comprises of external links to stream and/or snapshot
LiDAR	<missing>	-	
Human Location	Device <partial match>	https://smart-data-models.github.io/dataModel.Device/Device/examples/example.json	
Personal Schedule	<missing>	-	

Table 5.2-3: Human Digital Twin: Smart Data Models analysis

Data	Smart Data Models	Link	Comments
Wearable	<missing>	-	
MRI	<missing>	-	
Image	<missing>	-	

Medical data	ContinuousGlucoseMeterCalibrate ContinuousGlucoseMeterSamplingInterval ContinuousGlucoseMeterSensor ContinuousGlucoseMeterStatus ContinuousGlucoseMeterThreshold GlucoseCarb GlucoseHealth GlucoseMeal GlucoseMedication GlucoseSampleLocation GlucoseTester Glucose heartRateZone HeartRate	https://github.com/smart-data-models/dataModel.OCF/tree/master/ContinuousGlucoseMeterCalibrate https://github.com/smart-data-models/dataModel.OCF/tree/master/HeartRate	At moment available only for Heart Rate and Glucose
Weather	WeatherAlert WeatherForecast WeatherObserved	https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherAlert https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherForecast https://github.com/smart-data-models/dataModel.Weather/tree/master/WeatherObserved	

Table 5.2-4: Remote Robot Operation: Smart Data Models analysis

Data	Smart Data Models	Link	Comments
Plant	<missing>	-	
Area Map	GIS Data <partial match>	https://smart-data-models.github.io/dataModel.RiskManagement/GISData/examples/example.jsonld	This might be not a perfect match
Single object plants	<missing>	-	

Analysis result

The NGSI-LD meta-model and cross-domain model well suit the IOWN Digital Twin data requirements. However, as we can see from the tables above, there are two gaps for the domain-specific model:

- DTF-Gap-3: Overlapping data models. Data models from different domains sometimes overlap with each other. Needed to have an **ontology matching** to unify them
- DTF-Gap-4: missing data models. Not all the Digital Twin data from DTF use cases are covered by Smart Data Models. Needed to **specify new Smart Data Models**.

5.2.2.2. Context Broker Architecture

The Context Broker Architecture has been studied already into the IOWN Data Hub functional architecture [IOWN IDH]. The context broker is the core component that handles the data in FIWARE in the form of *context*. The implementation of the IOWN Digital Twin use cases comprises the following aspects:

1. within the single stakeholder there are multiple sources of data
2. the data management for a single stakeholder should be distributed into the edge-cloud continuum
3. there are multiple stakeholders to be federated

The architecture of the FIWARE context broker is very versatile. First possible architecture is Centralized architecture. The architecture has a centralized broker that handles all the data push from *context producers* and data requests from the *context consumers*. The data requests can follow two paradigms: query/response and subscribe/notify. With this architecture we can address the first Digital Twin requirement.

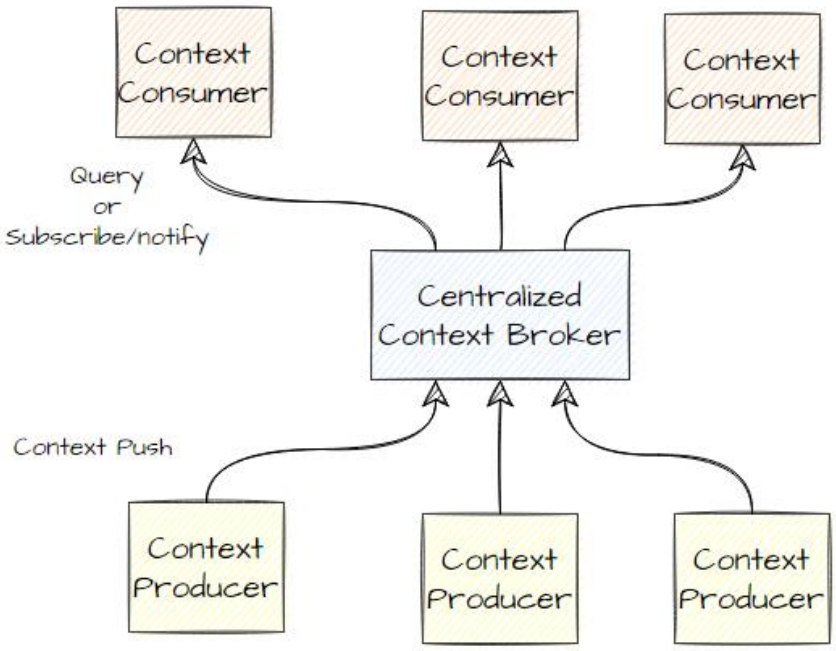

Figure 5.2-1: Centralized configuration of the context broker

A second possible architecture is a distributed architecture. Multiple context brokers handle data from different sets of context producers. The data is stored locally at each of the context brokers. A data request from the context consumer is dispatched by the *distribution context broker* to the local context broker(s) that hold the needed data. In this example of architecture we see two tiers of context brokers, that are the distribution context broker tier and the local context brokers tier. However, the number of tiers of context brokers can be arbitrary as it is needed. This kind of architecture suits the needs for distributed data management between edge and cloud.

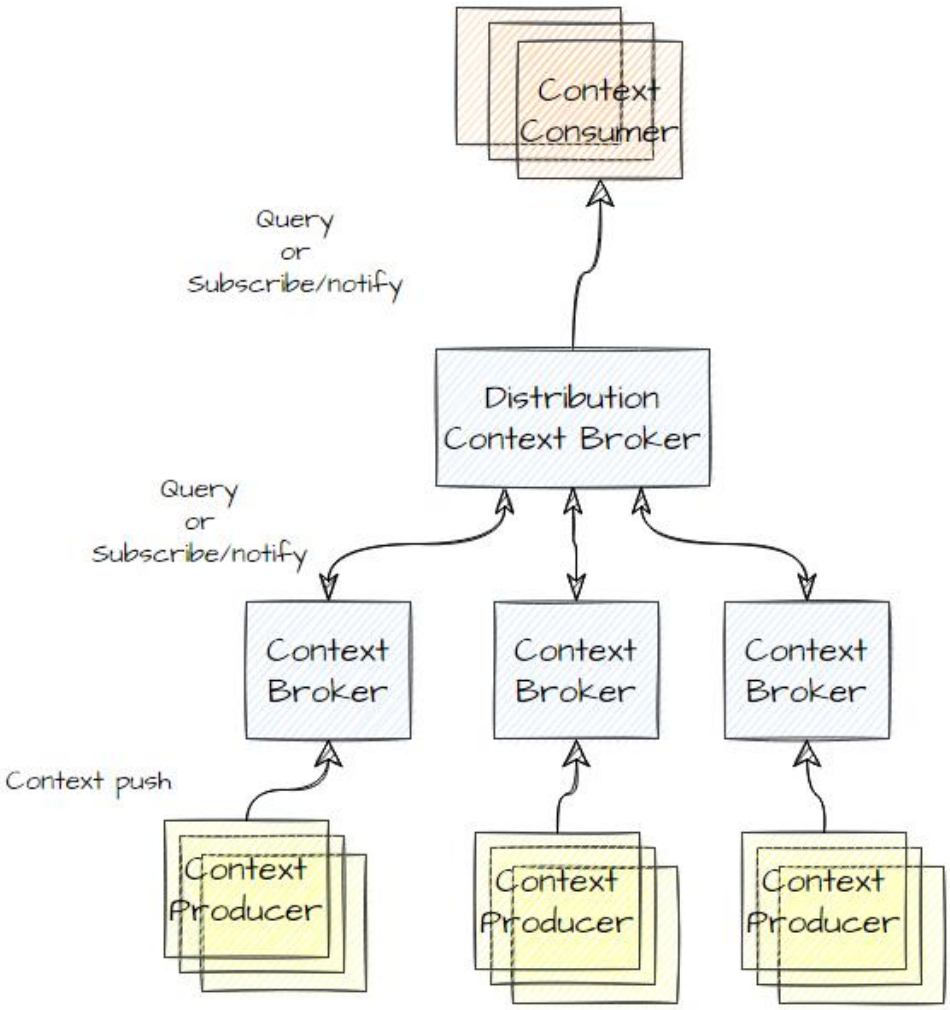

Figure 5.2-2: Distributed configuration of the context broker

The FIWARE context broker can be also hybrid such as the figure below. Different data stakeholders can maintain their own FIWARE context broker architecture and they can be federated by a *federation context broker.* A federation context broker is a distribution Broker that federates information from multiple underlying NGSI-LD Context Brokers and across domains [ETSI GS CIM 009]. This architecture suits the requirement for having a federation of data management.

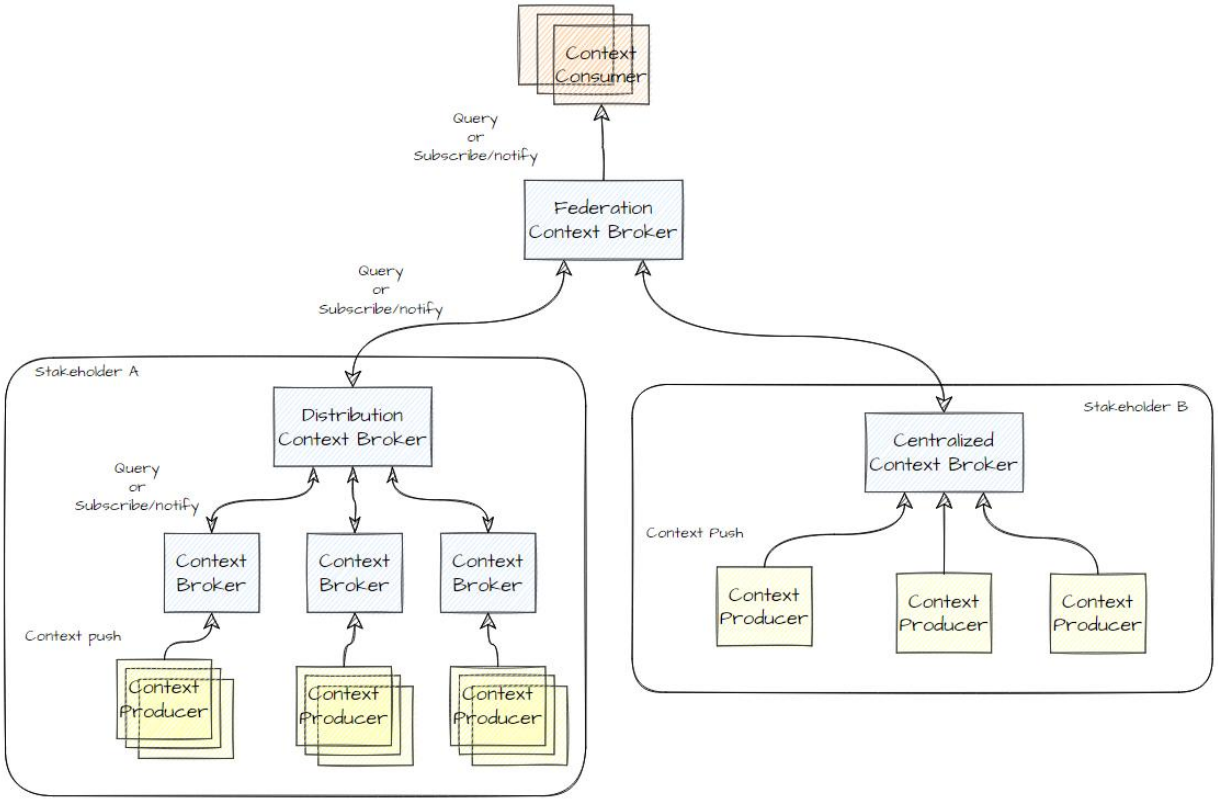

Figure 5.2-3: Federated configuration of the context broker

Analysis result

After this analysis, we can confirm that there are no gaps for the Digital Twin data management.

5.2.2.3. Digital Twin analytics

A digital twin is not only data but also analytics. Analytics process continuously operates on data to produce predictions of current, future or hypothetical status, and reaction to the status.

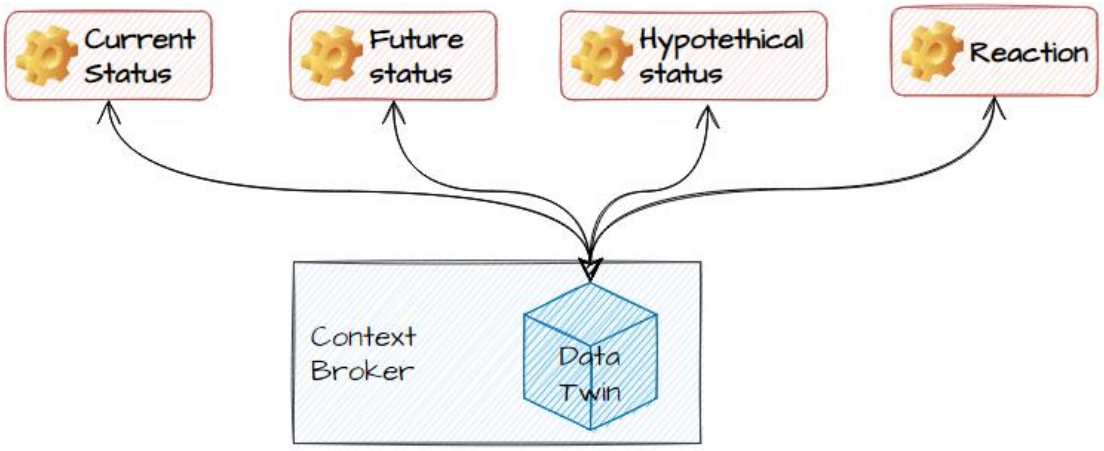

Figure 5.2-4: Analytics of Digital Twin

Analysis result

Currently FIWARE does not have a defined way to attach analytics to data, although a task force on this point is ongoing within the ETSI ISG CIM group based on use cases.

- DTF-Gap-5: **Linking Digital twin simulation models to Digital Twin data**. Currently ETSI CIM has a task force to study and address this gap.

5.2.2.4. Data analytics orchestration

Digital Twin analytics enhances the Digital Twin data continuously and new analytics based its processing on the inferred information. For example, in the case of the Green Twin use case, we might have that an Heating, ventilation, and air conditioning (HVAC) commander application, that issues command to the HVAC to minimize energy consumption and increase life quality of humans, takes leverage of crowd estimation from video camera, vehicle occupancy of public transportation based on Wi-Fi, and energy consumption prediction based on historical data. The design of this application is depicted on the top part of Figure 5.2-5.

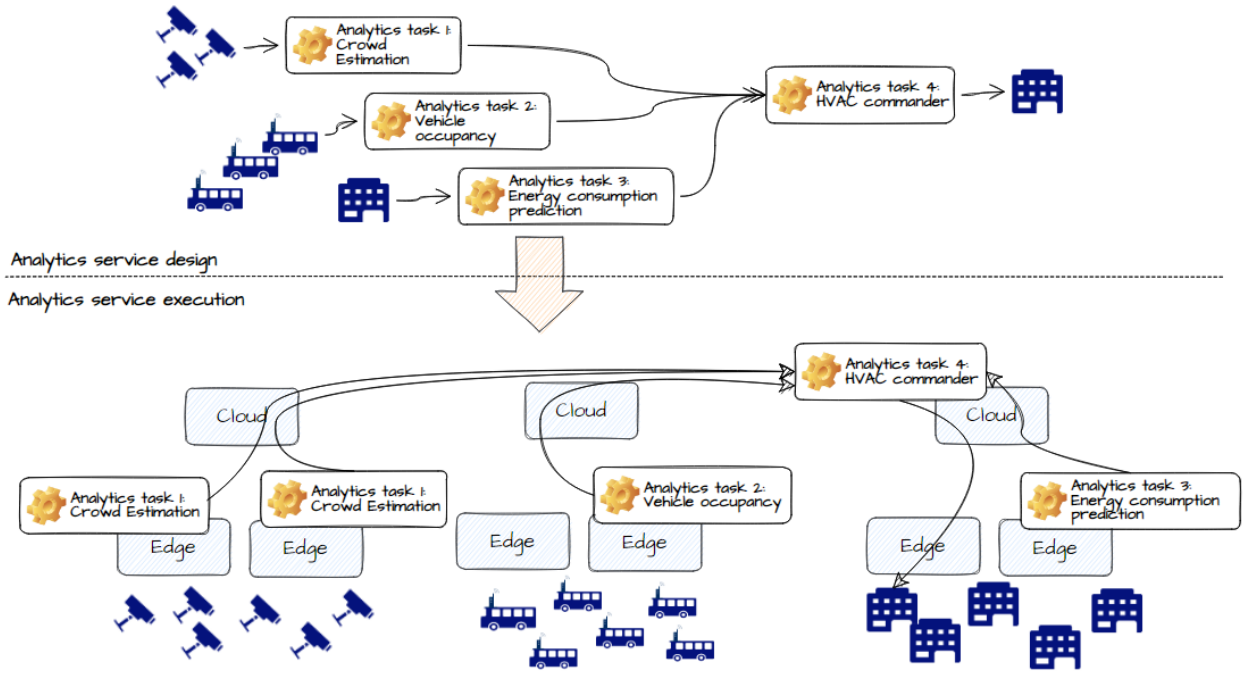

Figure 5.2-5: Difference between analytics service design and actual analytics service execution

However, the actual instance of the application might be much more complex. For example the computation might be distributed between edge and cloud, and shared among the federation of stakeholders. Some analytics tasks might be also replicated in multiple edges. A possible actual execution of the application is depicted on the bottom part of the picture above.

Taking into consideration the actual execution of tasks and establishment of the data flow might be a burden to a service consumer, especially for the use cases envisaged by IOWN GF.

A FIWARE system name FogFlow [FogFlow], [Cheng, 2017] automatically handles the execution of the analytics service starting from the analytics service design. FogFlow first decides on the number of instances of each task are needed depending on the available data, then it deploys the instances in the cloud and edge processing nodes. The current status of FogFlow foresees a centralized view of all processing nodes, thus, it is not yet federated.

Analysis result

FIWARE FogFlow is a good candidate for the orchestration of data analytics but there is still a gap for the orchestration among the federation:

- DTF-Gap-6: **Federated orchestrate data analytics**. In a cross-domain (multiple stakeholders) and distributed (cloud-edge continuum) infrastructure the orchestration of data analytics processes instances should be decoupled from the service definition and automatized

5.2.2.5. Data Usage Control

IOWN GF Digital Twin use cases foresee extended infrastructure deployment, distributed and federated as we have seen in section 5.2.2.2. Additionally, the data handled and processed by application is of big volume and velocity. The federation might involve quite a number of stakeholders.

Nevertheless, different stakeholders intend to keep the sovereignty and control over their data at any time during the data processing. Data usage control, thus, is a must for the IOWN GF Digital Twin use cases.

The current approach for data usage control is the *legal enforcement* such as for open data or for legal agreements between two parties. A data producer gives access to a trusted data consumer with a legal contract to constrain the usage of data. Whether the constraints of the data are applied is under full control of the data consumer. If there is an infringement, the data providers first need to discover it and then to act legally.

In this section we focus on about the *technical enforcement* of data usage control. Data usage control comprises both the access rights control and the usage control once the data is accessed (for example, anonymize before the usage).

The enforcement can be reactive or proactive. In the reactive approach if a misbehavior happens there is a reaction (e.g., the consumer application execution is stopped). In the proactive approach the misbehavior is avoided by algorithmic logic.

We need to consider for these two aspects:

- Data usage control might affect too much the data exchange overhead
- The configuration complexity to maintain analytics compliant and interoperable to different stakeholders policies might be hard to handle

The data usage control happens usually in two phases: the *configuration phase* done by data consumers and data producers, and the *enforcement phase* done by the technical system.

The configuration phase is when the data consumers and data producers configure the usage of data and protection of data respectively. In the first case, the data consumers configure the application to use data. In the second case, the data producers configure policies to protect data. For the configuration phase we analyze the configuration complexity. Measuring the overhead is not possible since there is no data flowing yet and no application running yet.

For the enforcement phase we analyze the overhead of enforcing the configurations. In this phase, the configurations are set and it is time for the system to apply them to enforce the data usage control. At this point, there is no intervention of data consumers or data producers, but only the technical system to run the algorithmic logic of the enforcement on the running applications and the established data flows. Thus, we do not analyze the configuration complexity but the overhead of enforcing the data usage control that might affect the application execution.

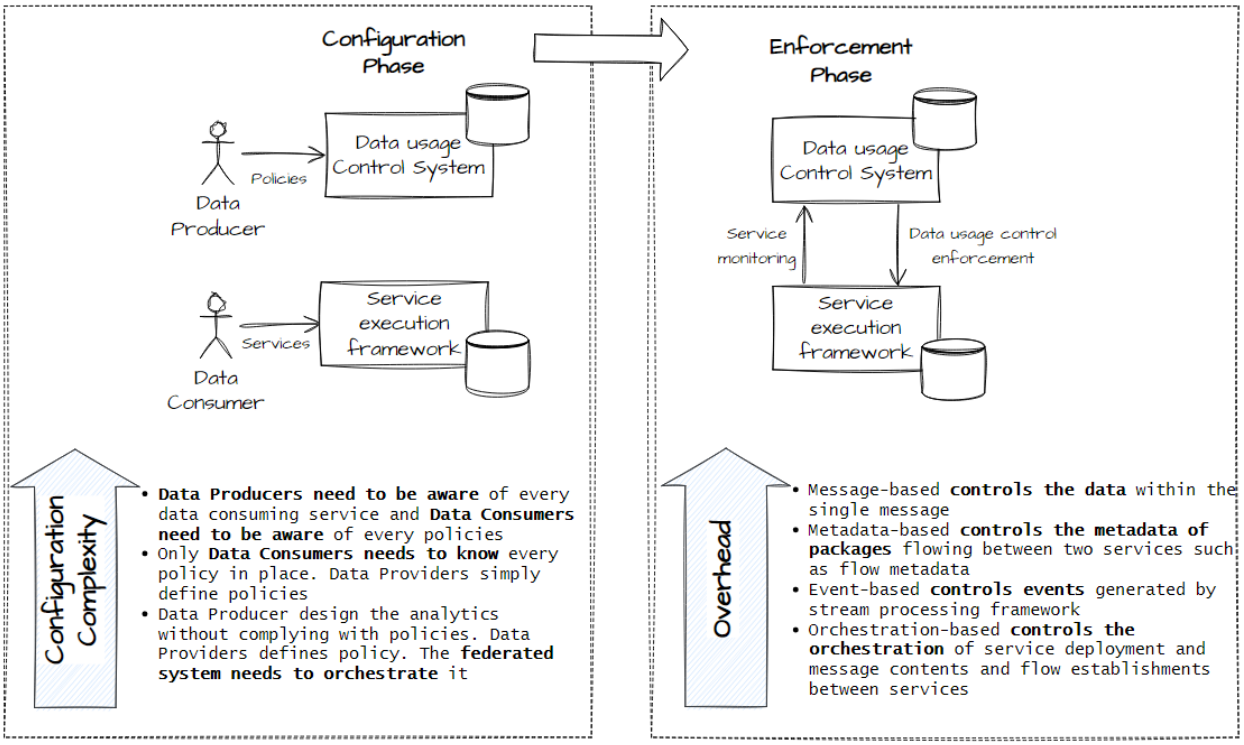

Figure 5.2-6: Configuration phase and enforcement phase issues in multi-stakeholders and big scale digital twin use cases

During the configuration phase, the case where the configuration complexity is the highest foresees that both the data producers and data consumers need to be aware of each other. The data consumer configures the data consumer service with the establishment of data flows and for the instantiation of the application processes. In order to be compliant with data usage control policies, the data consumer needs to be aware of the policies that affects the data to be consumed. If the policies are not respected, a reactive approach might shutdown the running services or a proactive approach might simply hinder the normal execution of the application. In some cases, the data producers must know the actual application processes that are running. A policy might allow a specific application instance of a specific data consumer to use the data, while might not allow the use of data for a different application of the same data consumer. In this case, the data producer is requested to know the actual application instances and the different data consumers.

In a case with less configuration complexity, the data usage control system checks that the application instance and the data flow are compliant with the policies. In case of detected misbehavior, a reactive system might stop the execution of the data consumer service, while a proactive system would hinder the execution of the data consumer service.

In the case with less configuration burden for data producers and consumers, the system is automatically embedding/reflecting the policies on the data consumer service. For example, the data usage control system might alter a data consumer service by automatically adding a pre-processing task, so that the data is first passed through the pre-processing task and then forwarded to the data consumer service. In this case, the data consumers do not need to know the data producers policies, but rely on the data usage control logic to automatically enforce them while ensuring the correct execution of the data consumer service.

During the enforcement phase, a data usage control system might implement the following approaches:

- Message-based approach checks the contents of every message exchanged between data producers and data consumers, and between sub-processes of every data consumer service

- Metadata-based approach checks the metadata that each of the message produces. This produces less overhead of the message-based because there is no need to parse the message. The number of checks is the same as message-based but each check is lighter than message-based.
- Event-based approach is mainly log based. It is not necessary that an event is generated for every message although that might happen (depends on the system settings). The events are then controlled asynchronously, that means that the message continues its path to the recipient while the events are analyzed. In case of misbehavior an action is taken. With this approach, the overhead is smaller (in general, less events than message-based) and the latency is smaller. However, there is possibility of data leakage for a short period.
- Orchestration-based approach targets the establishment of the data flow and the instantiation of the data analytics topology of the application. That implies that once the checks are made at orchestration planning, and the orchestration plan executed, there are no more controls to be enforced. In this approach the data consumers cannot interfere with the orchestration planning.

There are currently two research prototypes of data usage control in FIWARE: FIWARE Usage Control (despite the name it is not yet official part of FIWARE) and IntentKeeper. FIWARE Usage Control is based on the FIWARE security components such as Keyrock and Wilma [FIWARE Usage Control], [Munoz-Arcentales, 2020]. This data usage control system is event-based with a low complexity of configuration. IntentKeeper (Cirillo, Flavio, et al. "Intentkeeper: Intent-oriented data usage control for federated data analytics." *2020 IEEE 45th Conference on Local Computer Networks (LCN)*. IEEE, 2020.) is based on the FIWARE analytics orchestration component named FogFlow. This other data usage control system is orchestration-based. To configure it a data producer specifies the data usage control policies following a model requires only to specify the data types to control and the requested enforcement type. Other parameters, such as targeted data consumer ids and targeted data consumer service type, are optional. The data consumer, instead, designs its service type without considering the data usage control policies. The IntentKeeper system, then, matches the policies to data consumer services and alter the services design and the service execution to comply with the policies. The IntentKeeper configuration, thus, can be considered of low complexity.

Analysis result

Although the two prototypes from FIWARE are addressing some of the challenges of low overhead and low configuration complexity, a further analysis on the Digital Twin use cases implementation regarding the data usage control is needed.

- DTF-Gap-7: Less **overhead** for the enforcement of data usage control and less **configuration burden** for the data consumers and data providers

5.2.3. Identified Gaps

- DTF-Gap-3: Overlapping data models. Data models from different domains sometimes overlap with each other. Needed to have an **ontology matching** to unify them
- DTF-Gap-4: missing data models. Not all the Digital Twin data from DTF use cases are covered by Smart Data Models. Needed to **specify new Smart Data Models**.
- DTF-Gap-5: **Linking Digital twin simulation models to Digital Twin data**. Currently ETSI CIM has a task force to study and address this gap.
- DTF-Gap-6: **Federated orchestrate data analytics**. In a cross-domain (multiple stakeholders) and distributed (cloud-edge continuum) infrastructure the orchestration of data analytics processes instances should be decoupled from the service definition and automatized
- DTF-Gap-7: Less **overhead** for the enforcement of data usage control and less **configuration burden** for the data consumers and data providers

5.2.4. Conclusion

The analysis of FIWARE as technologies for interoperability and data usage control resulted in the identification of five gaps related to data interoperability, digital twin analytics services, and data usage control systems. Solutions to fill those gaps might be found already into experimental prototypes, research, and ongoing developments in the standardization activities. Industrial solutions based on FIWARE, such as the Thing'in platform that is based on the NGSI-LD standard (adopted by FIWARE) and described in the Appendix I.5, might address some of the identified gaps. In the future development of digital twin framework, IOWN GF will address those gaps in order to cope with the heterogeneity of digital twins, heterogeneity of stakeholders, and scale of the targeted use cases.

6. Conclusion

This document provides the use case analysis for digital twin as the first step of the Digital Twin Framework task force, and identified 6 requirements shown in Table 5.2-1. In these requirements, the DTF-Req-1 and DTF-Req-2 will be solved through collaboration with other IOWN GF WGs/TFs as they are non-functional requirements for network and computation infrastructures' capabilities and need to be defined based on actual use cases. On the other hand, other requirements (from DTF-Req-3 to DTF-Req-6) are functional requirements to support cross-industry exchanges of digital twins that are used in target applications defined in IOWN GF. Therefore, the gap analysis for these four requirements have been conducted.

Table 5.2-1: The requirements from use case analysis

No.	Requirements
DTF-Req-1	The IOWN GF infrastructure, including APN and DCI, is required to be flexible to accommodate various types of use cases that have critical differences of static and dynamic composition of digital twins.
DTF-Req-2	The DTF needs to find a way to define static and dynamic composition of digital twins based on use case scenarios and preconditions to identify concrete requirements for the IOWN GF infrastructures.
DTF-Req-3	The IOWN GF infrastructure needs to provide a platform to manage data flows of digital twins between different stakeholders. This requires access right management between stakeholders in terms of confidentiality, privacy, and efficiency of data exchange.
DTF-Req-4	The IOWN GF needs to provide a mechanism to manage confidentiality and privacy of different parts of one digital twin. This could be a guideline for structure of a digital twin with explicit indication of confidentiality and could need to collaborate with other SDOs.
DTF-Req-5	The IOWN GF needs to identify an interoperability mechanism to interact different types of digital twins. This could require collaboration with other SDOs.
DTF-Req-6	The IOWN GF needs to provide a data usage control system to control how the data is used after it is accessed by a data consumer.

In the gap analysis between the four requirements and the existing relevant technologies, 7 gaps are identified from access control policy management to data models shown in Table 5.2-2. Regarding DTF-Gap-1, 2, and 7, they relate to access control policy and enforcement operation of data usage control. These gaps need to be solved to implement cross-industry exchange of digital twins that are necessary functions to implement use cases IOWN GF aims to realize. Since there are several SDOs (e.g., FIWARE and Gaia-X) having similar discussions and experimental functions, these gaps are expected to be solved through a collaboration with external organizations led by the Digital Twin Framework task force.

On the other hand, DTF-Gap-3, 4, and 5 strongly relate to data models and interface defined in communities that are publicly accessible. The data model and interface issue of digital twins must be solved to support interoperability among necessary digital twins that are used together to implement a desired application. If we can’t guarantee the interoperability between data models and interface, other IOWN GF defined functions such as IOWN Data Hub do not work effectively in terms of applications' behaviors. A framework to use existing data models with guaranteed interoperability will be discussed in the Digital Twin Framework task force.

Finally, DTF-Gap-6 will be added to the work items of the Digital Twin Framework task force to find solutions for the federated orchestration issue through collaboration with other task forces in the IOWN GF as it affects architecture discussion of the IOWN GF infrastructures.

Table 5.2-2: The identified gaps from gap analysis

No.	Gaps
DTF-Gap-1	A mechanism to manage inherited access right policies that are created by previous stakeholders is not identified. As a digital twin contains many types of data added/modified by different stakeholders, accompanied access right policies need to be inherited from the original digital twin to the latest digital twin.
DTF-Gap-2	Policy assignment operation needs to be simplified. As a digital twin consists of many types of data that are consumed by different stakeholders, assigning and maintaining precise access right policies require huge cost.
DTF-Gap-3	Overlapping data models. Data models from different domains sometimes overlap with each other. Needed to have an ontology matching to unify them
DTF-Gap-4	Missing data models. Not all the Digital Twin data from DTF use cases are covered by Smart Data Models. Needed to specify new Smart Data Models.
DTF-Gap-5	Linking Digital twin simulation models to Digital Twin data. Currently ETSI CIM has a task force to study and address this gap.
DTF-Gap-6	Federated orchestrate data analytics. In a cross-domain (multiple stakeholders) and distributed (cloud-edge continuum) infrastructure the orchestration of data analytics processes instances should be decoupled from the service definition and automatized
DTF-Gap-7	Less overhead for the enforcement of data usage control and less configuration burden for the data consumers and data providers

The IOWN GF will continue to work on solving the gaps discovered through this analysis and will formalize solutions and guidelines by combining relevant technologies to use digital twins effectively in various types of use cases.

Appendix I: Relevant Technologies

I.1 FIWARE

The FIWARE is a curated framework of open source components to accelerate the development of smart solutions. In the last few years, FIWARE moved the focus of smart solutions mainly on digital twin applications. FIWARE is development-driven, that is applications and use cases in specific business domains are first studied and implemented and the components, expertise, models, and best practices are, then, uniformed and applied in other use cases. In the FIWARE vision, the new Digital Life gravitates around context data that describes what is going on, where, when, why, etc. Context creates a digital continuum, blurring the frontiers between application domains breaking the current silos of information.

FIWARE overall architecture is composable and it is already integrated with 3rd party platforms. The only component that must be part of a FIWARE-based solution is the Context Broker that is the software component that handles context data.

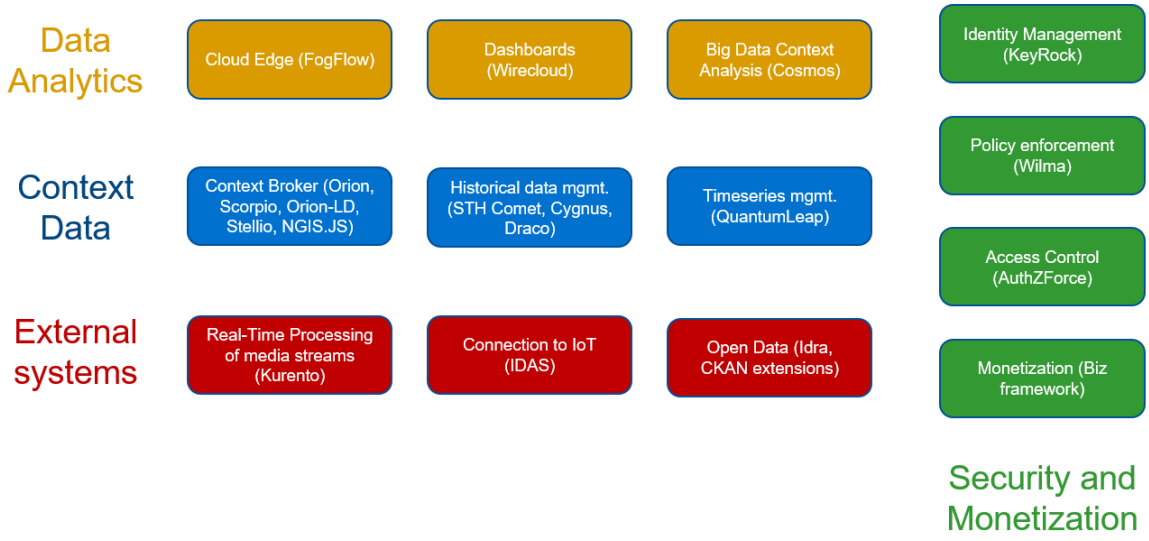

Figure I-7: FIWARE Components

Several business domains have been already studied as a whole by FIWARE. Reference architectures for smart city, smart factory, and smart agriculture have been defined by abstraction from specific FIWARE-based instances. Other user cases analyzed are smart energy and smart water.

Other results from the FIWARE community (in collaboration with IUDX, OASC and tmForum) are the smart data model that defines how to model data for different domains such street lightning, transportation, weather, parking, etc. Further, FIWARE community is closely collaborating with standardization bodies, such as ETSI, aiming at a standard interface for context information management (ETSI CIM). This new standardized interface is named NGSI-LD (Next Generation Service Interface - Linked Data). This work aims to achieve interoperability by the definition of common information models and interface. Smart data models are already specified in the same information model defined by ETSI CIM specification group.

I.2 USD

USD (Universal Scene Description) is a high-performance extensible software platform for collaboratively constructing animated 3D scenes, designed to meet the needs of large-scale film and visual effects production. And it is a system for encoding scalable, hierarchically organized, static and time-sampled data, for the primary purpose of interchanging and augmenting the data between cooperating digital content creation applications. USD provides robust interchange between digital content creation tools with its expanding set of schemas, covering domains like geometry, shading, lighting, and physics. USD's unique composition features have powerful benefits. For example, composition provides rich and varied ways to combine individual assets into larger assemblies, including asset and file references and variants, that let consumers aggregate multiple assets into a single scenegraph while still allowing for sparse overrides, and enables workflows that let many users collaborate simultaneously without conflict. And as a result of its power and versatility, it's being widely adopted, not only in the visual effects community, but also in architecture, design, robotics, manufacturing, and other disciplines.

USD is agnostic about the way material properties are represented. Artists should be able to author materials for cinema-quality rendering and have an automated process that produces simpler but still high-quality shaders in real time. To achieve this, NVIDIA has developed MDL, an open-source, GPU-friendly Material Definition Language with an associated distiller that simplifies shaders for preview and virtual reality (VR) applications. The NVIDIA MDL SDK has been adopted by many application developers. To facilitate its use, NVIDIA has created a specification (MDL Schema) for referencing MDL in USD and has developed Omniverse plug-ins to facilitate MDL-based workflows. MDL enjoys wide and increasing adoption in the design, visual effects, and gaming industries, among others.

Figure I-8: USD concept

I.3 DTDL

Digital Twin Definition Language, called DTDL, is the one of well-used frameworks to define accessible level of details for complex digital twins. Currently, as definition languages of digital twins, Data Definition Language (DDL), which is a generic language used to create and modify the structure of a data base objects, and Interface Definition Language (IDL), which is a generic language lets an object/program written in one language communicate with other programs regardless of a used language, are broadly known. DTDL, developed by Microsoft, is a language for describing models for IoT Plug and Play devices, device digital twins, and logical digital twins.

DTDL has two versions; the version 1 is designed for IoT hubs, and version 2 is designed for Azure Digital Twin. There exist similar technologies with DTDL v2, for example NGSI-LD developed by ETSI and Real Estate Core developed by Real Estate Core Project as a common language for controlling buildings.

The model used in DTDL v2 is similar to a class in an object-oriented programming language, and have names, such as “Floor” or “Room,” the elements, such as properties, and commands describing the type of entity. DTDL is based on defining a model with JSON-LD where there are four types of classes:

1. Interface (highest) class: “Interface” is a must for each digital twin model
2. Field class: Select 0 or multiple of “Property,” “Telemetry,” “Component,” “Relationship”
3. Element class
4. Schema class

DTDL is one of framework for describing digital twin. In order to efficiently exchange digital twin framework in future, standardization work on each detail level of the framework, shown in the figure below, needs to be considered.

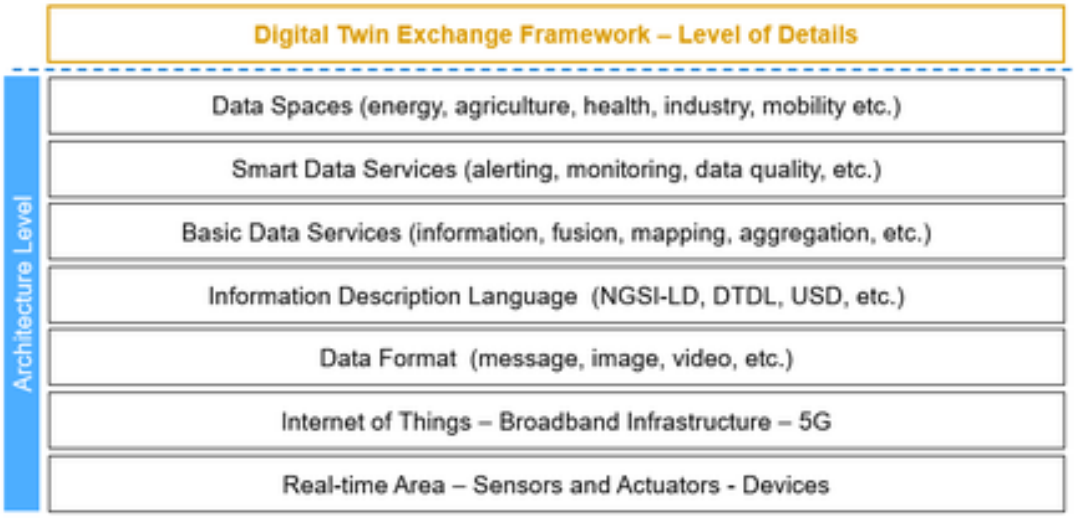

Figure I-9: The level of details of exchanging digital twin framework

I.4 GAIA-X/IDS

Gaia-X and IDS are projects that address the multi-stakeholder data sharing required by the digital twin. Gaia-X is an initiative announced by the German and French governments in October 2019 to support data sharing via infrastructure that ensures data protection, transparency, reliability, interoperability, and ultimately data sovereignty. It is an initiative to create an ecosystem of data spaces and has announced a system architecture using IDS technologies. In accordance with this concept, Catena-x has been launched by German automakers and suppliers for the secure distribution of parts information and other data among related companies.

IDS is an international architecture and standard for data spaces defined by the International Data Spaces Association (IDSA) and its members. It defines an IDS connector that provides access and usage control of data according to laws and contracts. The IDS architecture requires systems to communicate with each other using IDS Connector, which provides trusted communication between data providers and data consumers that are located in local devices, edges, and clouds. Communications of IDS Connectors are protected by Certification authority (CA) and Dynamic attribute provisioning service (DAPS), which issues tokens to verify dynamic attributes of IDS Connectors and system participants. IDS architecture defines basic concept of data access control and data usage control for maintaining data sovereignty.

I.5 Thing’in

Thing'in is a digital twin platform, it provides a place to model a cyber-physical environment, to create and manage Digital Twins. On the one hand, the context (the physical environment surrounding an object and the object’s relations

with this environment) is be modelled as a graph of digital twins provided by one stakeholder. The graph could contain the description of a building composed of rooms, walls, doors, windows, etc. or a router geolocated at a given position and interconnected to other devices, or anything else. On the other hand, Thing'in ensures data sharing between stakeholders, protection, reliability, interoperability, and ultimately data sovereignty. Other stakeholders could use any digital twin (or part of digital twin) if he has been granted. More information could be found at [Thinginthefuture].

Thing'in provides a complete mechanism to manage the security and the right for the different actors of the system. Thing'in through standard technology could interconnect with several identity providers and then identify and authenticate the stakeholders (through standard Identity Provider protocols like OauthV2 or OpenID Connect):

- with a Role Based Access Control (RBAC) can grant and tune the accesses to its API;
- with Access Control List (ACL) located at the level of a digital twin, the access to the digital twin could be ensured, this ACL could embed fine grain rules befitting to define access control on the properties of the digital twin.

Thing'in allows different levels of security (see [Thingin Security]), it will match easily with several use cases. The managed concepts useful to set up the security in Thing'in are:

- *User*: mainly defines the id of a stakeholder, and should be authenticate
- Role: a label, a User could have several roles
- *Resource*: mainly an API resource
- *Policy*: RBAC rules, defines the which *Resource* could be used by a *Role*
- *Digital Twin* (DT): a node the Thing'in graph, it represents the data/metadata of a physical object
- *ACL*: an access control list could be associated to a DT to manage the access to its properties. The access could be directly granted a *User* (via his id) or a *Role* to manage the whole DT (at DT level, all the DT properties are managed in the same way). Note: ACL can be mutualized between several DT.

The Thing'in core exposes, through APIs, digital twins as a graph (more specifically as a directed multigraph), i.e., a set of nodes (vertices, points) and directed links (edges) between nodes. Applications and services can create and manipulate these digital twins and the associated information (function, properties, state, location, shape, etc.), as well as the physical objects to which they are linked (e.g., thanks to *Access Modalities*, access to sensors and actuators and more generally to external platforms or repositories the digital twin is (inter)linked with), but also and above all the structural and semantic relationships between these objects. Thing'in core model is natively based on property graphs and implements the NGSI-LD standard [ETSI GS CIM 009] specified by ETSI.

Thing'in provides a common informational substrate that homogeneously describes states of the physical world (buildings, factories, cities, roads, and rails...) and therefore allows for the management of digital twins in different and possibly interconnected vertical domains (building, city, manufacturing, transport, etc.). Based on this informational substrate, more complex functions such as simulation or prediction can be developed.

From a functional point of view, Thing'in is based on two main components exposed through open APIs:

- The Core Graph managing the digital twin entities, their properties and relationships,
- The Model Dictionary, also named OLS (Ontology Lookup Service) managing the extensible catalog of ontologies, thesauri and vocabularies which can be used in Thing'in.

Ontologies and Ontology Lookup Service

Semantic capabilities in Thing'in are supported through the use of Web ontologies written in the Web Ontology Language (OWL). The Ontology Lookup Service (OLS), or ontologies server, aims to store and maintain ontologies used to provide semantic indexing and referencing of the entities which make up the core graph of Thing'in. Only ontologies present in OLS can be used to create, search, update, or delete. The incentive to store those ontologies in our platform is durability and stability. In fact, not all ontologies are always available online on the Internet. Sometimes there are not publicly accessible online or they may not be available all a given time. Storing ontologies and maintaining

their availability and durability inside the Thing'in platform allows to have more control on the stability of the platform by depending less on external services.

The other goals of the ontologies server is to provide search capabilities on ontologies concepts. That is, using a REST and/or GraphQL APIs, developers can download ontologies registered in the system and search among those ontologies, concepts to better describe semantically their data. Only a few ontologies, select and curated are registered in OLS. Developers can use the REST APIs to import new ontologies in the system. To ensure a certain level of integrity and/or quality we have a validation process on new ontologies before they can be used to index Thing'in entities.

Ontology prefixes management

Ontology namespace prefixes [W3C RDF 1.1] are managed transparently by the Ontology Lookup Service. Prefixes can be added by users if required through a dedicated API. These prefixes are used in the following formats for better human-readability and payload size reduction: Turtle, JSON-LD, TriG, N-Quads (see next section).

Data exchange formats

JSON formats

Thing'in provides support for json as an exchange format. This format includes two syntaxes including a compact syntax for large datasets. The json format includes all semantic information about entities (classes, object properties, data properties, annotations…), and graph-based meta-model properties (NGSI-LD), i.e., labels, relationships, and properties.

Detailed documentation:

- management: [Thingin AvatorCRUD]
- NGSI-LD standard integration: [Thingin CoreContext]

RDF syntaxes

Thing'in provides full compatibility with the RDF standard data format and all of its syntaxes. This compatibility is ensured both for Input and Output, meaning one can inject data already formatted in one of these formats, and request this data back in any format (including json formats). This includes any data injected by the means of injectors.

List of formats supported:

- TURTLE [W3C TURTLE]
- RDF/XML [W3C RDF 1.1 XML]
- N-TRIPLES [W3C RDF N-TRIPLE]
- N-QUADS [W3C RDF 1.1 N-QUADS]

  Thing in the future has a specific management of named graphs, see our dedicated guide: [Thingin Named Graphs]
- TriG [W3C RDF 1.1 TriG]

  Thing in the future has a specific management of named graphs, see our dedicated guide: [Thingin Named Graphs]
- JSON-LD [W3C JSON-LD 1.1]

  Thing in the future is compatible with JSON-LD 1.1 specification [W3C JSON-LD 1.1], including JSON-LD formatting algorithms in output.

Detailed documentation: [Thingin JSON-LD]

Additionally, experimental compatibility with W3C's recommendation RDF-star [W3C rdf-star] is proposed. This extension of RDF allows to include properties on edges, which in the context of Thing in the future allows for example to add and retrieve metadata about edges in the same manner as JSON formats (see [Thingin eJSONavator]).

Detailed documentation: Guide for RDF-STAR in Thing'in [Thingin RDF-STAR]

Geometric information management

Global management

WGS84 (GPS) is used as the main coordinate system in Thing'in. Geometric coordinates and queries account only for 2D. 2.5D (i.e., 2D+floor) in the context of buildings can be stored through semantic properties and queried/filtered. Other projections than WGS84 e.g., Mercator, Lambert, or indoor-specific references, can be managed through conversion to WGS84 (see Geometric conversions).

Supported geo formats:

- GeoJSON in JSON syntaxes
- GeoSPARQL-WKT in RDF syntaxes (OGC standard)

Geometric primitives, queries, operators

Thing'in query language includes geometric operators to perform filter on geometric primitives. These filters can be combined with other filters on attributes/properties of objects.

The geometric primitives include:

- Point
- Polygon
- LineString

The geometric operators include:

- geoWithin (Point, Polygon)
- geoContains (Polygon, Point)
- nearSphere (Point, radius)

Detailed documentation:

Search Features: [Thingin Avatars]

Visualization tools

Thing'in includes the following visualization enablers dedicated to geometric data:

- Google Maps
- Open Street Maps
- 2D plan (with image overlay). This viewer mainly targets indoor plans.

Geometric conversions

A user of Thing'in can define on-the-fly conversion of geographic data from a predefined projection (e.g., mercator, pseudo-mercator, Lambert 2, OSGB 1936 / British National Grid...). Such a conversion will convert geographic data from the projection format to the WGS84 (i.e., GPS) projection. It is possible to select whichever origin projection, but the target will always be WGS84.

Detailed documentation: [Thingin Manipulate]

Definitions and Abbreviations

Definitions

For the purposes of this Reference Document, the following definitions apply:

Term	Definition
Virtual Space	Virtually created space on computing system to accommodate various types of digital twins. Although there are several types of virtual spaces created by different software platform, it basically has three dimensional coordinates, functions to allocate digital twins in their virtual space with necessary meta data (e.g., identifier, location, and behavior), and interfaces to other systems to control / monitor the virtual space. For examples, a traffic simulator provides a virtual space as coordinates with locations of cars and pedestrians, and a game engine provides a virtual space as dynamic three dimensional space for characters and environments such as building, foliage, and vehicles.
Open Area use case	The Open Area use case is a kind of platform of smart community, smart city, and smart environment where many stakeholders share their data and knowledge each other to create a common data infrastructure based on open data.
Closed Area use case	The Closed Area use case is created for specific purpose with limited number of stakeholders.

Abbreviations and acronyms

For the purposes of this Reference Document, the following abbreviations and acronyms apply:

2D: Two dimensional

3D: Three dimensional

AI: Artificial Intelligence

AIC: AI-Integrated Communication

AMS: Area Management Security

APN: All photonic network

CCC: Central Control Center

CPS: Cyber physical system

DTDL: Digital Twin Definition Language

DTF: Digital Twin Framework Task Force

E2E: End to end

HVAC: Heating, Ventilation, and Air Conditioning

IOWN GF: IOWN Global Forum

RIM: Reference Implementation Model

References

[Cheng, 2017]: Cheng, B., Solmaz, G., Cirillo, F., Kovacs, E., Terasawa, K., & Kitazawa, A. (2017). FogFlow: Easy programming of IoT services over cloud and edges for smart cities. *IEEE Internet of Things journal*, *5*(2), 696-707.

[ETSI GS CIM 009]: ETSI GS CIM 009 V1.6.1 (2022-08) Cross-cutting Context Information Management (CIM); NGSI-LD API

[FIWARE Usage Control]: FIWARE Usage Control

[FogFlow] FIWARE FogFlow Documentation

[IOWN GF AIC] AI-Integrated Communications Use Case Interim Report, Version 2.0

[IOWN GF CPS] Cyber-Physical System Use Case Interim Report, Version 2.0

[IOWN IDH] Data Hub Functional Architecture, Version 1.0

[IOWN RIM AM] Reference Implementation Model (RIM) for the Area Management Security Use Case

[Munoz-Arcentales, 2020] Munoz-Arcentales, Andres, et al. "Data Usage and Access Control in Industrial Data Spaces: Implementation Using FIWARE." *Sustainability* 12.9 (2020): 3885.

[Smart Data Models] Smart Data ModelsVersion 1.0

[UniMurcia Facultad Medicina] University of Murcia, Facultad de Medicina, Map and Directions

[Thinginthefuture] Thing in the future Wiki

[Thingin Security] Thing in the future Wiki - Security and Confidentiality

[W3C RDF 1.1] RDF 1.1 Concepts and Abstract Syntax

[Thingin AvatorCRUD] Thing in the future Wiki - Avatar CRUD with JSON

[Thingin CoreContext] Thing in the future Wiki - Core Context

[W3C TURTLE] W3C RDF 1.1 Turtle Terse RDF Triple Language

[W3C RDF 1.1 XML] W3C RDF 1.1 XML Syntax

[W3C RDF N-TRIPLE] W3C RDF 1.1 N-Triples A line-based syntax for an RDF graph

[W3C RDF N-QUADS] W3C RDF 1.1 N-Quads A line-based syntax for RDF datasets

[Thingin Named Graphs] Thing in the future Wiki - Named Graphs

[W3C RDF 1.1 TriG] W3C RDF 1.1 TriG RDF Dataset Language

[W3C JSON-LD 1.1] W3C JSON-LD 1.1 A JSON-based Serialization for Linked Data

[Thingin JSON-LD] Thing in the future Wiki - JSON-LD support

[W3C rdf-star] W3C RDF-star and SPARQL-star

[Thingin eJSONavator] Thing in the future Wiki - Evolution of the JSON format of an avatar

[Thingin RDF-STAR] Thing in the future Wiki - RDF-STAR support

[Thingin Avatars] Thing in the future Wiki - Avatars - Search

[Thingin Manipulate] Thing in the future Wiki - Manipulate conversion functions by domain

History

Revision	Release Date	Summary of Changes
1	February 2023	Initial Release