



HAL
open science

Digital Twin and the Information Problem

Roberto Minerva

► **To cite this version:**

Roberto Minerva. Digital Twin and the Information Problem. Computer Science [cs]. Sorbonne University, 2024. tel-04854096

HAL Id: tel-04854096

<https://theses.hal.science/tel-04854096v1>

Submitted on 23 Dec 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial - NoDerivatives 4.0 International License

MÉMOIRE POUR L'OBTENTION DE L'HABILITATION À DIRIGER DES RECHERCHES

Sorbonne Université

Spécialité
Sciences de l'Information

Présentée par

ROBERTO MINERVA

Maître de Conférences,
Laboratoire S@movar,
Telecom SudParis

Digital Twin and the Information Problem

soutenue le 18 Decembre 2024
devant le jury composé de :

LUIGI ATZORI	FULL PROFESSOR AT UNIVERSITA DI CAGLIARI, ITALIE	Rapporteur
ROCH GLITHO	FULL PROFESSOR À CONCORDIA UNIVERSITY, CANADA	Rapporteur
STEFANO SECCI	PROFESSOR À CNAM, PARIS, FRANCE	Rapporteur
GUY PUJOLLE	EMERITUS PROFESSOR À SORBONNE UNIVERSITÉ, PARIS, FRANCE	Examineur / President
GIANCARLO FORTINO	FULL PROFESSOR À UNIVERSITA DI CALABRIA, ITALIE	Examineur
CHARALAMPOS PATRIKAKIS	FULL PROFESSOR À UNIVERSITY OF WEST ATTICA, GREECE	Examineur
ANTONIO SKARMETA	FULL PROFESSOR À UNIVERSITY OF MURCIA, ESPAGNE	Examineur

CONTENTS

Table of Contents	iii
List of Figures	vi
List of Tables	vii
Résumé	1
1 Problem Definition	3
Problem Statement - In The Large	3
1.1 The Context	4
1.2 Problem Statement: DT Modeling and Information	5
1.2.1 The Frame of Reference	5
1.2.2 The Fragmentation of DT Solutions	6
1.2.3 Towards a Generic Digital Twin Architecture	6
1.3 How and Why This Research Path Started	7
1.4 Research Context	8
1.5 Motivations and Needs	8
1.6 Purposes and Objectives	8
I State of the art	9
2 Definition of the Digital Twin	11
2.1 What a Digital Twin Is	13
2.2 Modeling Capabilities for a DT Definition	17
2.3 Digital Twin at Work	18
2.3.1 DT Modeling Practices	19
2.3.2 Digital Twin and Artificial Intelligence	19
2.3.3 Frameworks, Platforms, Tools for DT Development	20
3 General Purpose Sensing	21
3.1 General Purpose Sensing: Problem Statement	22
3.1.1 Definition of General Purpose and Synthetic Sensing	23
3.1.2 Applying General Purpose Sensing to Large Environments	24
3.1.3 The Role of Data Fusion to Support Synthetic Sensing	25
3.2 Digital Twin and General Purpose Sensing	27
3.2.1 The Problem Statement	29
3.2.2 Needed Technologies and Functionalities	30
II Contributions	31
4 Digital Twin Evolution	33
4.1 Architectural Framework	34

4.2	Modeling Problem	36
4.2.1	Descriptive Modeling	36
4.2.2	Prescriptive Modeling	40
4.2.3	Multi-Facet Digital Twin Problem	41
4.3	How Modeling Fits Into Architectural Framework	41
5	General Purpose Sensing: The Way Forward	43
5.1	General Purpose Sensing at work	44
5.2	Generating Data at the Edge	45
5.2.1	Opportunistic Digital Twin	46
5.2.2	Looking for Bottom Up Models	47
5.3	Reusing Data in a Smart City	48
5.3.1	Traffic Intensity and Pollution	48
5.3.2	Traffic Intensity from Google Data	49
5.3.3	From Redundancy to Optimization	50
5.4	An Architecture for General Purpose Sensing	52
III	Something New	55
6	Future Research Paths	57
6.1	Bottom Up Digital Twins	58
6.2	Digital Twin and Explainable AI	58
6.3	Situation Awareness: Back to Reasoning	59
6.4	How Much Information from Sensing	60
6.5	A Possible Roadmap	61
	Bibliography	79
IV	Annex	81
A	Research Background	83
		83
A.1	Current Research Topics	84
A.2	More on the Modeling Problem	90
A.3	Current Prescriptive Templates	92
A.3.1	Aggregation of Cooperating Objects	92
A.3.2	Goal-based independent Objects operating in the same Environment	94
A.3.3	Open System Monitoring and Optimization	95
A.3.4	Scenario based Digital Twins	96
A.3.5	Multi-facet Modeling	100
A.4	Research over the Years	102
A.4.1	Preamble	102
A.4.2	Research Fields	102
A.4.3	Oh, That Book	107

A.4.4	Open Contributions	107
A.5	Additional Information	108
A.5.1	Teaching	108
A.5.2	A Bit of Metrics	108
A.5.3	Mentoring of Students	110
A.6	Acknowledgments	110

LIST OF FIGURES

1.1	Architecture with Integrated Modeling.	6
1.2	Modeling Integration for Lifecycle.	7
2.1	DT simplified Lifecycle	16
2.2	DT Platform: from Ideation to Execution	18
2.3	DT Modeling Example	19
2.4	AI usage during different phases of the DT life cycle	20
2.5	Different Modeling Capabilities needed by a DT and their intertwining	20
3.1	Correlation Analysis of Traffic Intensity, Pollutants and Meteo Factors	25
3.2	LSTM Features and Hyper Parameters for Traffic Pollution Prediction	26
3.3	Comparison of MAE Values for Data Fusion Input and only Traffic Intensity.	26
3.4	Organization of Noise centered Experiments	27
3.5	MAE for Noise - Traffic Analysis.	27
3.6	Signature as a Bridge for General Purpose, Synthetic Sensing and Digital Twin.	28
3.7	From "Normal Situation" Construction to Anomaly Detection.	28
3.8	An architectural View for Bottom Up Construction of Digital Twin.	29
4.1	A Reference Architecture for DT Applications based on FIWARE.	35
4.2	Madrid and Dublin Data Structures for Traffic Intensity	37
4.3	Network Digital Twin Representation and Support for Stakeholders.	39
4.4	An integrated IDE for DT Implementation	42
5.1	General Purpose Sensing System for Traffic Intensity Detection from Noise Signatures.	44
5.2	GPS System for Signatures and DTs Creation.	46
5.3	General Purpose Sensing and Synthetic Sensing as Enablers of Opportunistic Digital Twin	47
5.4	Analysis of Madrid Data for correlating Traffic Intensity and Pollutants	49
5.5	Finding the Correlation between District 19 Sensors.	51
5.6	Correlation Map and LSTM Prediction Results for correlated Sensors	52
5.7	A Schematic Distributed Architecture for General Purpose and Synthetic Sensing	53
6.1	Research Roadmap for the identified Activities	62
6.2	Possible Exploitation Paths.	62
A.1	Interplay of descriptive and prescriptive Models	90
A.2	Interplay of different Modeling.	91
A.3	Aggregation of Objects with a common Goal	93
A.4	Representation of an Aggregation of Objects with a common Goal	93
A.5	Representation of Cooperating Objects within a Common Environment	94
A.6	Towards a formal Representation of cooperating Objects with Papyrus	95
A.7	Representation of a monitoring System	96
A.8	Climaborough Project: Two types of Digital Twin for the Smart City	97

A.9	Climaborough Project: generic Architecture for supporting different DTs	97
A.10	A simplified recycling Process	98
A.11	Problem Functions and possible Relations to actual Data	99
A.12	A DT Representation Of Bernini Bust of Louis XIV showing different Facets of its Representation and its possible Integration into different Systems	101
A.13	Analysis of Research Impact	109
A.14	Scholarometer Results and Relations Maps.	109

LIST OF TABLES

5.1	Comparison of Performance Metric for Predicted Models	50
5.2	LSTM Model Performances in terms of MAE, MSE and RMSE	50



RÉSUMÉ

Le jumeau numérique est étroitement associé à l'évolution des capacités de détection et d'actionnement, à la gestion des données, au big data et aux techniques d'intelligence artificielle, ainsi qu'aux plateformes de service pour le développement et l'exécution de grandes applications sur des systèmes de edge - cloud largement distribués.

Le concept permet également le développement de nombreuses applications dans divers domaines et secteurs, offrant souvent des avantages commerciaux significatifs. La mise en œuvre d'un jumeau numérique couvre différentes phases du cycle de vie d'un système et nécessite un soutien dédié de la part de diverses équipes et groupes au sein de l'organisation qui le développe ou l'utilise.

Le concept de base du jumeau numérique est la capacité de représenter, par le biais de logiciels, toutes les caractéristiques et comportements d'un objet ou système physique dans un environnement (opérationnel). Cette représentation repose sur la modélisation descriptive, qui consiste à définir et décrire des modèles de données englobant toutes les informations nécessaires au fonctionnement du jumeau numérique, et sur la modélisation prescriptive, qui consiste à décrire pleinement le comportement prévu et attendu du jumeau numérique, reflétant celui du système physique réel. La définition, le développement et l'exploitation d'un jumeau numérique nécessitent une quantité importante de données et l'association de ces données avec le comportement du système. Un effort de développement considérable est requis au fil du temps pour aboutir à une mise en œuvre correcte du jumeau numérique.

Ce document décrit plusieurs tentatives visant à faciliter la modélisation et le développement de solutions basées sur les jumeaux numériques à partir des modèles descriptifs existants, des outils de modélisation prescriptive, de la définition de modèles de comportement, ainsi que de la capacité à combiner "general-purpose and synthetic sensing" pour accéder à toutes les données nécessaires pour soutenir une représentation numérique d'un système physique. La détection à usage général fait référence à la capacité d'utiliser des capteurs de manière à capturer autant d'informations que possible de l'environnement et à appliquer des technologies d'intelligence artificielle pour transformer ces données en informations exploitables. Cette phase de transformation rendue possible par l'IA est appelée détection synthétique pour indiquer que le contenu nouvellement généré a été inféré et n'a pas été directement mesuré sur le terrain.

Le rapport explore les voies possibles et les conséquences de l'adoption de ces technologies pour le développement de jumeaux numériques, ainsi que la nécessité d'étendre les plateformes disponibles pour soutenir cette approche.

PROBLEM DEFINITION

The beginning is the most important part of the work.”

– Plato

1.1	The Context	4
1.2	Problem Statement: DT Modeling and Information	5
1.2.1	The Frame of Reference	5
1.2.2	The Fragmentation of DT Solutions	6
1.2.3	Towards a Generic Digital Twin Architecture	6
1.3	How and Why This Research Path Started	7
1.4	Research Context	8
1.5	Motivations and Needs	8
1.6	Purposes and Objectives	8

The thesis describes a research path towards the creation of an effective and actionable software framework for the development of componentized Digital Twin, DT. This platform aims at exploiting large sets of raw data and information acquired in disparate manners and rationalized by means of AI. Two major topics will be addressed: *modeling of Digital Twin* and *the exploitation of General Purpose Sensing*. The combination of these two capabilities opens up a field of research originated from the need to better represent the behavioral aspects of DT and move to a larger use of general purpose sensing for feeding, through AI techniques, the information hungry DTs. For a description of the research and technical background of these activities, a historical perspective of concerned ICT fields is in Annex A. As readers might imagine, this research is not a linear, easy, nor comfortable path. The document describes some steps, failures, and attained results that have characterized this research endeavor. The approach aims at academic goals but with a pragmatic flavor due to a long industrial innovation experience. The document comprises different parts: a Problem Definition (this chapter) introduces the domain issues; Part I (with two chapters) comprises contributions that are consolidating; Part II (with other two chapters) describes current research efforts and the work carried out in strong cooperation with a number of students (Doctorands and Master students); Part III is providing hints on the shape of things to come; Part IV is an Annex. A.1 contains a list of Research Topics, RT, and a brief explanation of them. It is used along the document as a reference to Research Topics (referred to as RT.xx). Annex contains more information on "Modeling Problem of DT" (A.2) and "Behavioral Templates" (A.3). A.4 is a recap of the personal history of over 30 years of research. A.5 offers additional information about teaching, metrics of the research work, and management of students and their activities. A.6 contains acknowledgments to people that helped along this path.

I hope you will enjoy this journey as much as I did ...

1.1 The Context

Information and Communications Technology industry, ICT, is changing due to increasing power of software and Artificial Intelligence. A consequence is a pervading softwarization that affects every aspect of our daily life [1]. Software evolution is pushing towards componentization and distribution of functionalities [2] close to users, as the rise of Edge-Cloud Continuum is showing [3]. There is an increase in market size for applications (e.g., Internet of Things, IoT) that employ distributed processing, storage, communications and sensing/actuation [4]. Deployment of sensors and actuators is still pursued in a "vertical" manner, i.e., specialized devices are deployed to control a specific phenomenon. This leads to large sensing infrastructures requiring integration of several device types. An example, that will be reused throughout this document, is the city of Madrid [5]. In this way, infrastructure gets bigger, needs more management, and it is very heterogeneous. A new field of investigation (RT.6 and RT.7) is the use of general purpose sensing to infer synthetic data [6]. The approach has been proposed for small environments, and now it is extended to larger ones [7]. Another characterizing factor is a constant trend towards virtualization. The concept of Digital Twin, DT [8], is gaining interest beyond the Manufacturing Industry. DT enables high levels of softwarization and servitization [9], [10]. This thesis is addressing some issues, problems, and possibilities stemming from the use of Digital Twins exploiting information generated by "general purpose and synthetic sensing".

1.2 Problem Statement: DT Modeling and Information

The usage of DTs is spreading [11]. Successful solutions are sophisticated, complex, expensive, specific, often proprietary and pointing to a single environment or problem. They require a lot of know-how to be developed, maintained and used [12]. They need strong modeling foundations: descriptive (e.g., data models) and prescriptive (e.g., behavior definition)[13]. DT development frameworks are mainly supporting descriptive modeling, while prescriptive ones are left to developers (RT.3) [14]. Another issue (RT.13) is optimization of IoT platforms and a better formalization of information fed into DTs during different steps of their lifecycle. The injection chain [15] for transforming data into information should be transformed into a fabric capable of producing the needed information at the right time and in a standardized form for the DT to use in different phases and contexts. The complexity of data transformation should be transparent to the DT. This separation of concerns helps in design the DT by using well formed descriptive data models and provides a layering principle for DT development frameworks. DT development comprises several issues:

- The prescriptive modeling and the support to developers for more easily creation of DT applications (RT.3). Currently Model Based System Engineering, MBSE[16], supports design and implementation of large systems from scratch in a top-down fashion. It is necessary to ease the development by providing templates of DT components to be specialized by developers (RT.16).
- How to represent the behavior of an existing system to be implemented as a DT (RT.4). The joint action of modeling capabilities plays a major role [17]: the improvement in data modeling, the accuracy of prediction and the incremental construction of a prescriptive model can lead to a bottom-up system's representation. A methodology, supported by guidelines, tools, and frameworks to be used for incrementally building and consolidating bottom-up DT solutions, is necessary.
- Explainable DT. The interplay between Prescriptive and Predictive Modeling in a DT can help in explaining why a DT is performing certain actions to reach the predicted states . This research (RT.8) is at its early stage [18], [19].
- Coordination between simulation and execution (RT.11). Simulations and execution use different engines, the prescriptive model they use is adapted to these engines. Modifications can lead to slightly different models and, on a long run (if code modifications for simulation and execution are not coordinated), the two models can diverge. Keeping a unified prescriptive model, interchangeably usable for simulation, execution, and corroborated by predicted data, can be an major value of a DT.

The problem statement of this research can be enunciated as: *How to support and integrate the different models needed to describe a large system as a Digital Twin in such a way to democratize and make easier the development of DT based solutions? Is it possible to develop Digital Twin frameworks exploiting general purpose and synthetic sensing?*

1.2.1 The Frame of Reference

DT applications development is based on a vertical approach pursued by a specific organization addressing a well-determined problem. This results in the construction of a specialized system modeled during different phases of its lifecycle. These models are often specific for problem domain, industry and sometimes for enterprise [11]. There is an interest for representing systems of diverse problem domains (from smart cities to e-health) beyond the manufacturing one. There is also a need to bypass limits and boundaries imposed by spe-

cific industries. This requires optimization in aggregating sensed data; availability of interoperable data models, or ontologies; access to well-formed behavioral models, multi-facet integration for digitized systems (RT.22); an open and reliable development architecture that integrates modeling, simulation and prediction of future system’s states.

1.2.2 The Fragmentation of DT Solutions

Particularization is due to requirement on DT applications: peculiar goals in application’s construction; high specialization of the system to be represented; high cost of development, modification/improvement, and management of the DT solution, and competitive advantage of well-formed models [8][20]. For these reasons, available software solutions are focusing on data definition, while the behavioral modeling is very specific to the problem domain at hand [21]. Availability of Data Models [22] or the definition of metalanguages for representing DT, as Microsoft’s DT Definition Language[23], are defining a common ground for "data understanding" and interoperability between applications [24]. It is necessary to recompose the fragmentation providing an homogeneous capability to operate on well formed information (by means of programmable injection chains, cooperation between general purpose and synthetic sensing, wide scope definition of ontologies and data models); and the identification of behavioral templates supported by reusable components to be specialized to substantiate specific problems. This points to functional separations between data, behavioral, and simulation/predictive layers.

1.2.3 Towards a Generic Digital Twin Architecture

The outcome is a horizontal DT architecture capable of supporting the three modeling facets in an integrated manner. It is based on the ability to collect generic data, readily transform them into actionable ones, and use them to describe the behavior of represented system or parts. Simulation and prediction features should be supported, with prescriptive modeling providing actual evidence of the system behavior and explaining predicted, simulated results. In addition, the simulation and execution engines should be integrated in such a way that any change to the behavior should be immediately accounted for and actuated by both engines.

A reference functional architecture is represented in Fig.1.1. It aims at the integration and interplay of different modeling capabilities and is complemented by a fabric for data transformation based on synthetic sensing (pushing for a large scale adoption of general purpose sensing). This is a componentized architecture, whose elements can be distributed and exploit capabilities of edge-cloud continuum. Data can be locally acquired and processed at the edge and only relevant information forwarded to cloud components. This is ideal for general purpose and synthetic sensing. Digital Twin is a concept re-

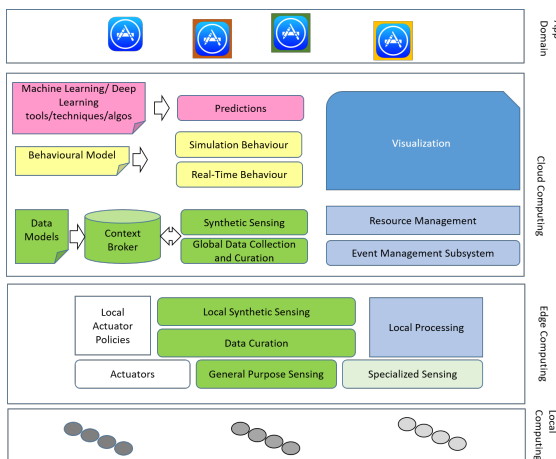


Figure 1.1: Architecture with Integrated Modeling.

nents. This is ideal for general purpose and synthetic sensing. Digital Twin is a concept re-

lated to system’s lifecycle, therefore the whole architecture should support different phases and, in each of them, the modeling integration. Relationships between a prescriptive model in one phase (e.g., design phase) with the one of another step (e.g., operational stage) is an important research issue that should be tackled (but it will not be further discussed in this document). For the time being, two phases are briefly treated in the document. During

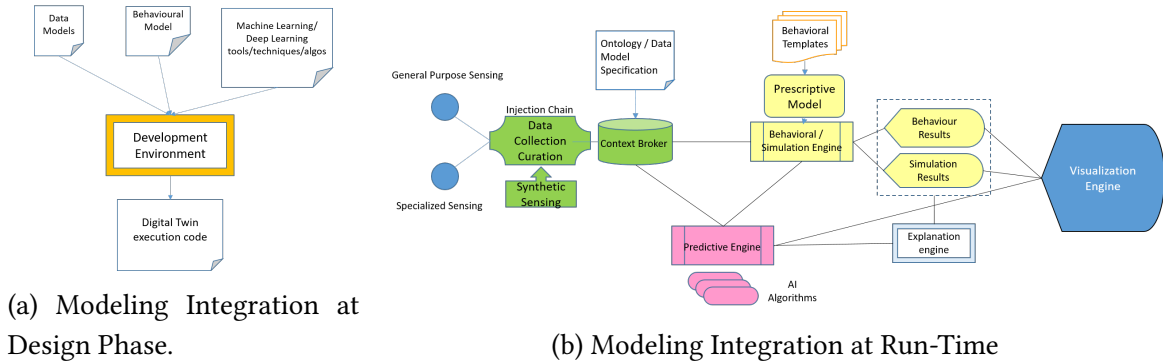


Figure 1.2: Modeling Integration for Lifecycle.

the *design phase*, the goal is to specify and prototype features of the DT. The combination of data, planned behavior, predictability of future states should be considered, designed, developed and tested all together. In Fig.1.2a, these aspects are integrated. There is not a fully fledged DT development environment. There are a few candidates based on tools supporting Model Based Design Engineering [25] or devoted to mathematical modeling [26]. The paper [27] tries to integrate different modeling techniques to harmonize prescriptive modeling (top-down) with descriptive one based on data. There is a need to provide contributions to this research endeavor. At run-time, a DT makes use of several functionalities that need interoperability of data and synchronization of events and actions during distributed execution. Fig.1.2b represents different aspects and tools needed to make a DT implementation effective.

1.3 How and Why This Research Path Started

The research about Smart City application development was focusing on the possibility to predict evolution of traffic intensity within Madrid with ML/DL techniques [28]. A subsequent step was to study if and how data fusion could improve prediction results. Fusing different open data sets (traffic intensity, pollutants in nearby locations, meteorological status), the accuracy of prediction was higher than just using the traffic intensity data set [29]. These encouraging outcomes drove to the idea of verifying if a General Purpose Sensing approach [6] applied to larger environments represented as Digital Twins could hold [7]. Actually the goal was to figure out if traffic intensity could be inferred by considering other parameters like pollution and noise intensity. A set of experiments encompassing implementation and small deployment of General Purpose with extensive Synthetic Sensing capabilities supported by AI technologies (discussed in next chapters) was set up to prove the hypothesis. A parallel research thread was pursued: the digital twin representation of complex objects within Internet of Things applications [30], [31]. They also achieved results (further discussed in next chapters). The two trends of research concocted into the idea that

a DT architecture could be the right solution to frame capabilities of general purpose and synthetic sensing into an operational framework capable of reconciling data (descriptive), predictive and prescriptive models. Initial work on the usage of DT with general purpose sensing was addressing computational aspects and the needed infrastructure [7], [32].

1.4 Research Context

The technical topics and subjects touched by this thesis are many and they cover different fields of ICT. It is difficult to provide a clear and linear line of evolution of the entire set of addressed research topics. During the research, hypotheses failures, new hints, and ideas stemming from different technological fields have created a path that follows the most promising directions and accommodated skills and interests of involved researchers. Some basic fields of research and knowledge are:

1. IoT architectures and Sensing/Actuation frameworks.
2. Edge, distributed systems and their integration with Cloud resources.
3. DT representation and modeling of complex objects operating in an environment.
4. Software and service architectures and large scale componentization.
5. Usage and improvement of Artificial Intelligence tools and techniques for manipulating data and create/infer reliable information.

1.5 Motivations and Needs

Some software platforms, e.g., [22], [23], [33], claim their ability to support DT development. Each platform has merits, but the development of an effective general DT framework is a long process with lots of implications on the physical system/object lifecycle, on enterprises processes and on the supporting infrastructure [20]. For the development of a well-structured and effective Digital Twin, more than Data Description Metalanguages and well organized repositories for accessing structured and linked data are necessary. There is the need for a fully fledged infrastructure that can democratize the development of complex DT applications by means of integrated descriptive, prescriptive and predictive modeling and simulation, and a set of well defined components and their Application Programming Interfaces, APIs. The availability of a DT - IDE can enlarge the number of developers and practitioners that adopt a DT based approach and foster the sharing of new components, data models and predictive capabilities cross fertilizing different domains.

1.6 Purposes and Objectives

The purpose of this research is, then, the definition of a general software architecture for the creation of Digital Twins of large systems capable of serving applications and functionalities for controlling a physical system. The goal is to remain "General Purpose" and create an horizontal framework usable in disparate application domains. Emphasis is also on the possibility of validating the "general purpose sensing" capabilities within large and complex real-world environments. This effort aims at integrating the modeling requirements for an effective development of DTs by using the most valid and powerful tools, and solutions existing in the open source domain. However, the attempt is to demonstrate the possibilities and not necessarily to develop a Digital Twin development environment.

Part I

State of the art

DEFINITION OF THE DIGITAL TWIN

The way we experience the world around us is a direct reflection of the world within us

– Gabrielle Bernstein

2.1	What a Digital Twin Is	13
2.2	Modeling Capabilities for a DT Definition	17
2.3	Digital Twin at Work	18
2.3.1	DT Modeling Practices	19
2.3.2	Digital Twin and Artificial Intelligence	19
2.3.3	Frameworks, Platforms, Tools for DT Development	20

Despite its long history, the Digital Twin, DT, has been defined, contextualized, and popularized within the manufacturing industry. It is used as a software representation of a physical object / product within different periods of its lifecycle as clearly clarified by the work and perspective of Grieves [8]. From Negroponte's idea of moving "from atoms to bits" [34] back in 1995, initial virtualization of physical objects were proposed in different areas: intelligent agents [35], multimedia and eXtended Reality [36], also IoT domain [37] [10]. There exist many definitions of the Digital Twin [38], they share a common view of a data based software entity that represents essential characteristics of a physical object. Other features, characterizations or definitions of DT slightly diverge depending on specific perspectives or different problem domains. As stated in [8] and [12], the Digital Twin offers the opportunity of aggregating and organizing data around a physical object's model¹. This means collecting a large number of data for describing the dynamics of physical object's behavior and, by exploiting models or Artificial Intelligence techniques, to understand and predict the future behavior[39]. Although this apparent simplicity, the definition, implementation and usage of DTs come with relevant challenges and require meaningful efforts to bear the expected benefits.

DT represents substantial knowledge and know-how of a problem domain: modeling it in an effective and exploitable manner is not a trivial task. A lot of effort has to go in the representation of behavior and characterizing aspects of the object by means of available data (RT.3). Behavior description is increasingly complicated for larger systems made of many interacting components. Descriptive modeling requires data identification, aggregation, curation, and manipulation that are relevant aspects of the DT formalization.

Quantity of data, especially in dynamic systems, can be overwhelming. Data are collected and processed in different systems and by means of many functions and algorithms. They can span over a very large distributed system. The optimization of algorithms, functions and processing/storage/communication capabilities can quickly become a problem. And data can be required in a timely fashion to fit the DT representation (RT.13).

Situation awareness is another DT challenge (RT.10). Key issues are the concepts of environment, context, and situation. To the best of author's knowledge, the applicable definitions are [40](very old) and [41] (newer) that relate situation awareness to a context and its status. From a very basic perspective, the DT definition needs to lay on a more detailed formulation of these concepts. The usage of ML/DL is sometimes of no avail because used data sets typically do not comprise data related to critical or rarely occurring situations. So associating Situation Awareness to ML/DL capabilities not necessarily is a good approach. In an enterprise, the DT will be used by different groups with specific objectives and representation requisites. This creates further complexity and requires a clear segmentation and adaptation/flexibility, synchronization, and attribution of responsibilities to internal business processes and a clear definition of contexts.

The challenges briefly introduced provide an explanation about the difficulties in developing, using and maintaining successful DT implementations. However, in spite of all these challenges, the DT development has reached an inflection point [42], and its exploitation is becoming more usual in many organizations. For a detailed description of current state of the art, from definition to major technical challenges and a plethora of appealing application cases, the readers can refer to the book [11] and A.4.3.

¹During an editing meeting for the book [11], Micheal Grieves stated that: "*If you have enough data about a physical object, then you can create a Digital Twin of that entity*".

2.1 What a Digital Twin Is

The previous concepts support a basic understanding of the Digital Twin. An additional advancement is needed: the DT should be better characterized with respect to software architectures and platforms, i.e., the definition of a software architecture supporting the creation of DT based applications (RT.1); and conceptualization should move from vertical and application domain specific definitions to a general one supporting broader concepts underlying the Digital Twin. A first step is towards a software oriented definition, then facets of DT should be further analyzed and related to software properties. With the previous conceptualizations in mind, it is possible to start from the definition provided in [30].

Basic Definition of Digital Twin: *the constant association between a **complex object**² and its software representations in a virtualized environment along the object's lifecycle. Events, actions, features, status, data and other information characterizing the object are timely reflected by the logical (software) objects (and vice versa) within the specific environment.*

This is a high level definition, it should be further characterized by expressing some properties, features, and relationships that better characterize the complex object and its twin relations. Due to the diverse natures of objects and their possible representations, a large number of properties can be considered. Actually, the choice of a relevant set of main properties that characterize the Digital Twin is a major challenge in the definition of a working and effective Digital Twin. A primary property is that the complex object and its components are identifiable and their identities are associated to addressable software entities (each one with its identity) comprising the Digital Twin. Components' identity is fundamental because it guarantees the correct association between different parties. In addition, it can provide security, privacy, ownership features and relevant added value services [43], [44]. Some basic properties are expressed in the following and adapted from [30] and [31]:

- *Modeling and Representativeness.* This property enforces the definition of a model capable of describing the complex object's and its components behaviors and salient characteristics (as well goals and constraints). The model should be able to describe the object behavior over time on an environment.
- *Environment modeling and contextualization.* This property requires the definition of a model representing specific environment(s) in which the complex object operates.
- *Reflection.* This property dictates that the Digital Twin represents the current status of the complex object. The complex object's status (and its components) is exactly reflected/mirrored by the Digital Twin.
- *Entanglement.* This property states that data exchange from complex object to DTs takes place as soon as possible, i.e., in a suitable time that satisfies users' requirements and the intended usage of the DT functions within applications. Entanglement can be: unidirectional, the data flow from the complex object to the DT enabling reflection and other properties; bidirectional, the DT timely receives the data needed to reflect the object's status, but the DT can enforce states or behavior changes directly on the complex object or on some of its parts. In the first case, the physical object is not modifiable, in the latter one, it can be changed and "controlled" by its DT.

²Disclaimer: the term complex object is preferred to physical object, PO, because it points to an entity of aggregated components with behavioral complexity. A complex object is a synonym of system or aggregation of physical objects. However, to avoid misinterpretation between "complex objects" and "complex systems", this definition prefers the term complex object, but PO, system will be used interchangeably.

These are considered basic properties because they describe fundamental aspects of a Digital Twin: modeling and data reflection, and how they take place. Note that, for complex objects or objects with relevant lifecycle shifts, these basic properties alone pose important challenges. The object's model and its operational environment can change during the lifecycle, e.g., the environment for testing could be substantially different from the operational one, because the testing will consider and stress extreme cases.

Some simplifications or adaptations can make sense and lead to weaker ruling of these properties. For very simple objects, the modeling could be just represented by a status information. Entanglement property could be strongly relaxed for object whose status seldom changes. Modeling of complex objects and their parts is mandatory, but, when the environment does not affect the complex object behavior, its modeling can be very simplified or even disregarded. The basic properties are usually found in a DT representation, but they are not mandatory. And this can also point to the ephemeral nature of the DT definition and its difficult implementation. There is another relevant set of properties that are related to the softwarization of complex objects. They describe how the DT can implement the physical object and imply some software capabilities that pose challenges proper of highly distributed systems. The software set of properties is as follows:

- *Persistence.* This property prescribes that the DT instances are persistent independently from the physical object capability to be present and active. The software representation should be always-on and, even when disconnected from the physical object, provide some basic functions (e.g., the latest known status, the attempted goals, latest activities).
- *Replication.* This property points to the possibility of replicating DT instances to serve a larger audience or for supporting particular needs of certain users. The basic data and information should be shared between replicas (subject to the CAP/Brewer Theorem [45] implications) in such a way to guarantee the correct usage of all the replicas' functions. Each replica can also provide a specialized view per client or problem, in an attempt to ease the development of different software applications exploiting DT capabilities.
- *Memorization.* This property requires the DT's to store and maintain all the previous states and relevant information over time. This properties guarantees that the DT keeps "memory" of its past [46]. Memorization is instrumental to the use of large data sets for understanding and foreseeing the future status changes of a physical object or for use in simulations.
- *Composability and Aggregation.* This property indicates the possibility to aggregate / compose DTs (or their components) into more complex and larger DTs. In other terms, it guarantees that a DT can be seen as a meaningful and consistent complex object made out of component DTs each one well formed and capable of representing an element / subsystem of the physical object or a larger system (RT.12).
- *Augmentation.* This property states that functionalities offered by a complex object can be augmented by DT software. The DT, as a software artifact, can have interfaces and manage data. It can implement new functions and define or extend an Application Programming Interface, API, for controlling and programming the physical object's behavior and data. New information and functions on existing data can be created to enrich the set of functionalities supported by the complex object.
- *Predictability and Simulation.* This property ensures that the DT and its historical data

can be used for predicting or simulating future states of the complex object depending on different starting conditions. Prediction and simulation can consider various time frames and applying / project the analysis (of data) to future or past. Varying the initial conditions in past occurrences can help to understand how the object would have behaved under particular circumstances (RT.11).

- *Accountability and management.* This properties states that the DT has mechanisms to guarantee its accountability. A DT has built-in functions to be managed and accounted for, and it should support management functionalities by providing data and related functions.

The memorization property, in many cases, is a large collection of time series and this offers the possibility to access to curated datasets. AI based applications can use them for prediction or generative tasks. Guaranteeing the consistency and synchronization of different components' representations is a daunting software challenge, but it is a needed one for the progress of DT basic technologies' use. The composition is currently one issue (RT.12, RT.14) that needs research efforts due to the possibilities it offers to incrementally create larger well-synchronized DTs. The combination of memorization and predictability / simulation results particularly useful for testing purposes or the analysis of less frequent situations and instances. Prediction, together with modeling and contextualization (RT.3), is a typical task that involves a lot of Artificial Intelligence technologies (Machine and Deep Learning) [39]. The management capability ensures the ability to set the stage for supporting the other properties and associated functionalities. For instance, entanglement, reflection, replication, persistence, memorization and others do depend on the ability to manage the DT software system in a proper way. In addition, it is important to keep track of DT's users and ensure security, and privacy in the access to its data and functions. On a longer term perspective, DT and its component can be capable of self-management, opening up, in conjunction with situation awareness capabilities, the possibility to Autonomic DT (RT.10). Other properties are related to usage and business perspectives of a DT specification and implementation. Example properties are (more can be considered):

- *Ownership.* This property states that each DT and its components should be associated to an owner. This property rules the responsibility and rights between a DT and its owner. These constraints/obligations could be defined in terms of smart contracts.
- *Servitization.* This property refers to the possibility of transforming products into services and in particular Digital Twin supported ones [47] [48]. This property represents how the product is transformed into a service by means of software implementation of a DT and under what constraints and assumptions.

The ownership of physical objects can be relatively straightforward, while the issue of "ownership" for software counter-parts is somehow more complex and newer. There is the need to understand the ownership and delegation between the physical object and the DT. In some cases (e.g., rental cars on the city streets), the DT can assign a temporary ownership to a replica associated to a specific owner (e.g., the renter) and the physical object. This property has plenty of social and economical implications, and can also have an impact on possible business models and applications. The servitization property is useful to identify the ways and mechanisms for supporting the transformation of physical objects into services and what DT's functions and data will fall under this specific governance. Augmentation, ownership servitization properties can be used to enable applications that transform complex objects into services under the control of DT representations.

The DT should accommodate users and applications needs, so other properties can be progressively identified and considered. For each specific DT definition and implementation, a set of "functional" properties should be chosen and pursued to guarantee that the DT representation works and provides the expected functionalities. The selected properties aim at defining and molding the DT's capabilities into a well understood and suitable execution framework. The selected properties should be observed and comply with requirements during a specific DT's lifecycle phase. A few ones are listed below as examples:

- *Responsiveness.* This property states that DT's response time shall satisfy applications' timing requirements. Invocations of functionalities, availability of data should be executed and guaranteed in due time so that applications can respect their stated Key Performance Indicators, KPIs.
- *Scalability.* This property guarantees that the DT, its components and functionalities can scale up with respect to the number of internal entities (e.g., further decomposition of the DT in sub-components) and external one (e.g., more applications requesting functions and data from the DT). Scalability is also related to the number of replicas that the DT can support in relation to the expected number of customers.

Additional properties can be envisaged and added for specific requisites and expected behavior of the DT. Some of them are: Context Flexibility, the capability of adapting the DT behavior to a slightly different application context; Situation Learning (RT.10), i.e., the ability to add rules, constraints and action depending on the specific "situation" the DT is in that moment; Bottom Up Modeling (RT.4), i.e., the ability to extract knowledge and behavior understanding from the analysis of data generated by the physical object.

The DT's properties are not a static and "cast in stone" set of prescriptions, rules and constraints defining how the DT should behave. There are no "mandatory" rules (even if identity and modeling are characterizing the DT). The set of chosen and implemented properties characterizes the representation of complex object. This set can be dynamic and representing the required capabilities in a specific moment of the lifecycle.

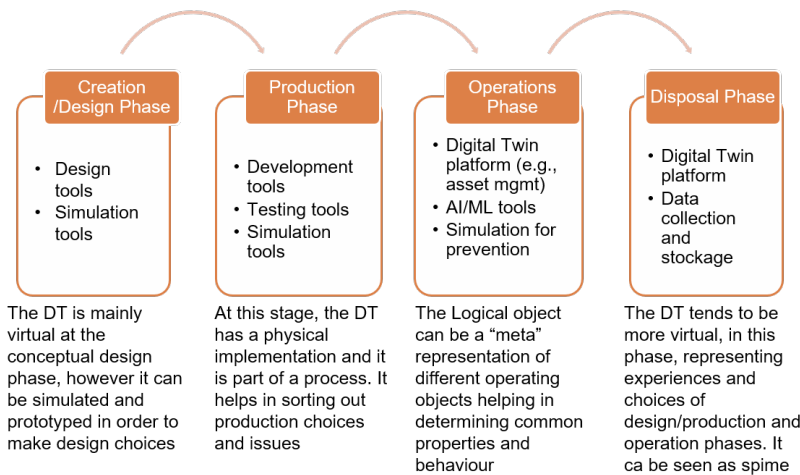


Figure 2.1: Simplified DT Lifecycle

In different phases, the properties can change and vary. New properties, requested by specific needs of the DT representation in that step, can be introduced. For instance, moving from testing to operation, a generalizing property can be requested/introduced, i.e., the DT represents all the general problems of all physical instances of the product (e.g., an aircraft can be represented by

a DT with all the issue reported about that specific model). The DT's behavior needs

to be very dynamic to grasp the complexity of complex object capabilities and actions. The properties are a means to identify a set of features that the DT will adhere to for a specific period of its lifecycle. The DT should be flexible and reflect the varying and changing characteristics as well as usage of the complex object. This is a major challenge in definition and implementation of DTs. As an help for the reader, a simplified lifecycle model for a DT is represented in Fig.2.1.

After this introduction to DT's properties, the basic DT definition can be improved along three main perspectives: one about the importance of contextualizing the object's behavior with respect to a target environment; one about the definition of software and functional properties for guiding the implementation of DT and platforms; another about the object framing in its different phases of the lifecycle with respect to software capabilities. A new definition of the Digital Twin can now be provided

Definition of Digital Twin: *the constant association between a complex object and its software representations within a virtualized environment. This representation holds during the object's lifecycle. Events, actions, features, status, data and other information characterizing the complex object and the environment in which it is operating are to be timely reflected by logical (software) objects (and vice versa) within the specific virtualized environment. The association between a complex physical object, PO, and DT, and between the Real Environment, RE, and the Virtual Environment, VE, is defined in terms of properties pertaining to the set of DT characterizing properties and they are supported by a DT software framework that helps in the implementation, execution and fulfillment of them.*

This definition enforces the definition of a virtual environment and the characterization of DT properties in relationship with a lifecycle phase. In the rest of this document, the discussion focus is on the design and creation phases (if not differently indicated).

2.2 Modeling Capabilities for a DT Definition

The business value of a Digital Twin is its capability to represent the complex physical object operating in a specific environment. The value lays in its effectiveness of providing actual status information, possible future behavior advice and in simulation of prospecting / potential actions considering initial conditions of the DT and VE. In industrial practice, the modeling of a product is carried out in two different facets: descriptive and prescriptive modeling [13]. In this approach, descriptive modeling deals with the definition, specification and implementation of "data models" that aim at describing in a well-formed and interoperable manner the data that characterize the DT. The subtlety is that a data model represents the DT changes of status over time, i.e., the results of the physical object actions (behavior) on the environment. A prescriptive model, instead, represents the course of actions, depending on initial states and external events as well as constraints and rules that govern the DT behavior and determine actions that results in DT's change of states³. The difference should be evident, the prescriptive model describes the course of action of the DT for reaching its goals (partial or global) in terms of events, actions, commands and reactions, and their possible sequencing; while the descriptive model represents the results of these actions, the format of events and commands as well as interfaces of the DT. A descriptive model may be handy for invoking DT functionalities (for instance by means of APIs[22] or through a Definition Language[23]). Under the push of increasing usage of Ma-

³it is important to note that the behaviors of complex object / PO, and DT / LO are considered equivalent and are used interchangeably.

chine / Deep Learning, ML/DL, techniques, a predictive modeling capability is available. It has created a mix up between descriptive, prescriptive, simulation and predictive modeling. The availability of large data sets (the memorization property of the DT) under the form of curated data has been exploited to predict future results of DT's actions and it has been related to DT behavior. To clarify the difference, a comparison between a DT with simulation capabilities executing a prescriptive model under current environmental conditions and a predictive engine that uses time series for determining the PO's status in the future can be useful (RT.11). On one side, there is the actual execution/simulation of intended actions upon receipt of events or commands, on the other side there is a prediction based on historical data. There are numerous differences in achieved results: the data set can be biased towards normal cases and hence the DT will not be able to correctly predict unexpected situations, while the DT simulation actually executes a prescriptive model representing the intended behavior of the complex object (RT.3) and, if initial system conditions are correct, it will provide accurate results also in rare cases (if the model was designed to do so). If the prescriptive model is effectively an actual reflection of PO's behavior, then its results are accurate and explanatory. An important advantage of the prescriptive model over the predictive one is, in fact, the ability to explain / provide evidence why certain actions have been taken and their consequences on the PO and environment (RT.8).

2.3 Digital Twin at Work

In articles [12][20], complexity and intricacies of DT usage, especially in large organizations, are discussed: data and models are central constituents of well-formed Digital Twins. The design, development, deployment, and operation are extremely complex steps of implementation (Design/Creation and Production Phases of the lifecycle) and they require an adequate support in terms of skills, teams, procedures and costs.

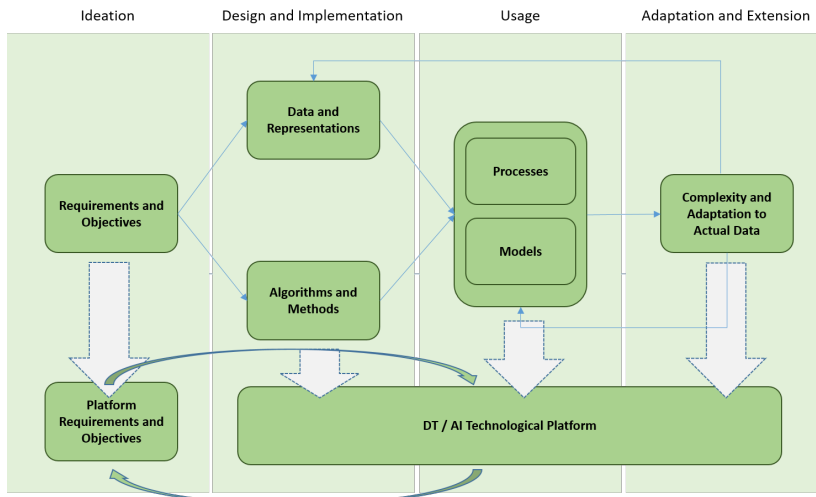


Figure 2.2: DT Platform: from Ideation to Execution

Fig.2.2 represents different steps: problem's conceptualization; identification of data, representations and algorithms, and methods to represent the system; implementation in terms of models and processes, that require a continuous adaptation to tune the models to actual data. The Digital Twin implementation is characterized by a continuous tuning up of data representations and models towards a

more precise alignment with actual data. This work requires skills and effective processes because it is expensive and time-consuming. There are essential major technical aspects to consider in these activities: the design of Digital Twin, the amount of data to deal with and to frame correctly into models, the intelligent algorithms and Artificial Intelligence

tools used to describe and assess the intended behavior, and the ability of the technological platform to support these substantial computing tasks. A few aspects of these themes are addressed in the following sections and are based on the results discussed in the book[11].

2.3.1 DT Modeling Practices

Fig.2.3 depicts salient aspects of the modeling process. Different data models and ontologies are needed with a prescriptive representation of the intended behavior of the object respect to the operational environment. In addition, management aspects, proper of a supporting platform, are to be considered for a fully effective execution of the DT. As discussed in

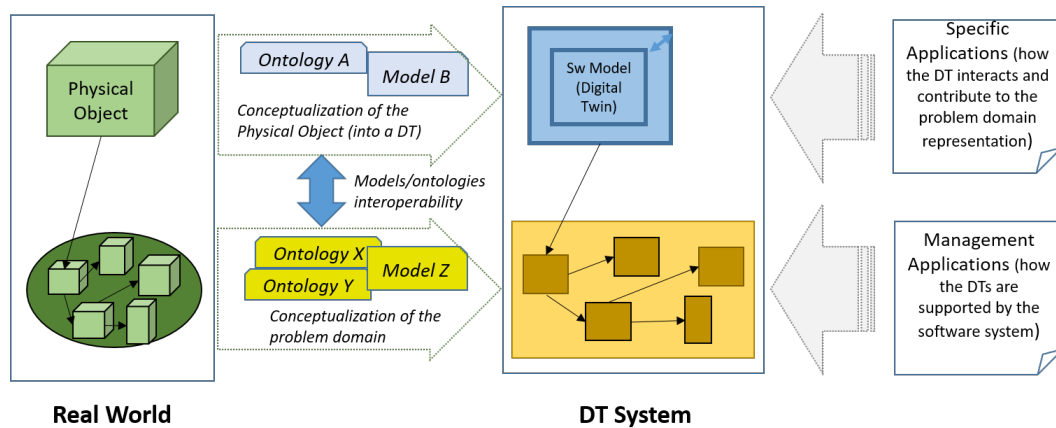


Figure 2.3: DT Modeling Example

[49], data modeling practices are more developed and supported by tools than prescriptive ones. The latter are left to developers because more related to a specific view and goals of the problem domain. It is necessary to reconcile the modeling aspects and to support the developers with well formed prescriptive models, components, and, possibly, tools for speeding up the developments.

2.3.2 Digital Twin and Artificial Intelligence

The usage of Artificial Intelligence techniques for supporting the development and execution of Digital Twins is a relevant trend. The paper [39] provides a systematization of different possibilities and a classification of DT depending on their usage of AI techniques⁴ in four types. The basic level is a **Passive DT** that simply reflects data collection to help in data analysis and visualization⁵. A **Predictive DT** uses prediction algorithms to anticipate the physical system states. **Reactive DT** and **Autonomous DT** are capable of suggesting or taking smart decisions that can lead to system behavior changes in some cases. Autonomous DT are capable of taking actions by reasoning on the situation to mitigate problems. As indicated in [39], AI can also be instrumental to help in different phases of the DT construction. Fig.2.4 depicts some possibilities. Generative AI can for instance be used to design critical

⁴Surprisingly, at the time of editing the paper, there were not too many surveys on this subject even if the combination of AI and DT is certainly a hype ...

⁵At first sight, this type of DT seems quite uncommon, but in reality it is widespread. The DT is mostly understood by general public as a 3D visual tool for representing what is going on in an environment. Visualization, in this document, is probably neglected, because it is considered complementary to more important aspects as modeling and reasoning.

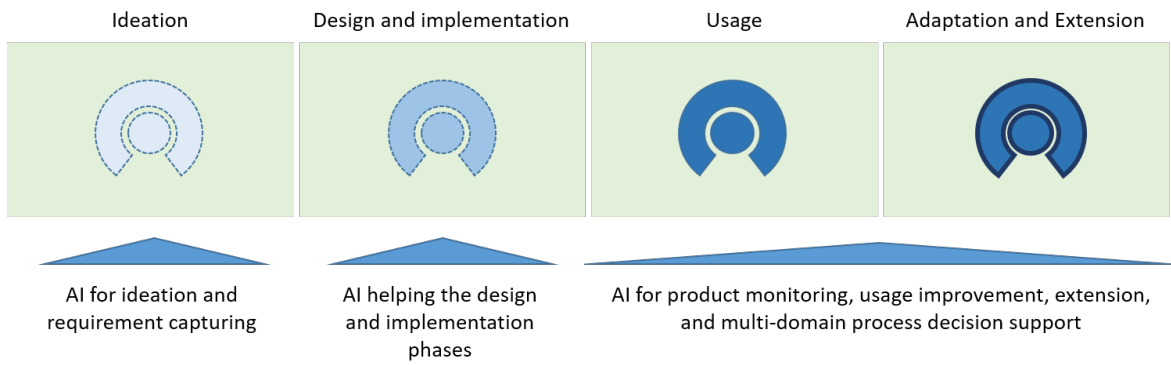


Figure 2.4: AI Usage during Phases of DT Lifecycle

situations or for testing the DT implementation. More research is needed to identify how AI can support the entire DT's lifecycle.

2.3.3 Frameworks, Platforms, Tools for DT Development

There are several frameworks and platforms supporting the DT development. Some notable solutions are: Azure DT from Microsoft [23] or GE's Predix⁶. There are also open source solutions like FIWARE [22] or Eclipse [33]. They essentially focus on data, either as representation and storing (e.g. FIWARE's Context Broker) or the support to data communication (e.g., Eclipse's DITTO). As discussed in [49], the main focus is on descriptive modeling with prescriptive representation left to programmers by means of tools and methodologies like MBSE [16], [25]. There are opportunities to further investigate how the different ways of modeling a system as a DT can be better supported and exploited by an Integrated Development Environment (RT.23).

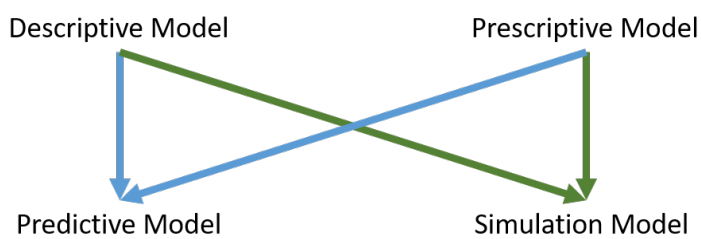


Figure 2.5: Different Modeling Capabilities needed by a DT and their Intertwining

Fig.2.5 shows different possibilities of interacting between descriptive (mainly data based) and prescriptive modeling (mainly physics, mathematics, heuristic formalization), prediction (by means of AI algorithms) and simulation (one of the major properties of the DT). Around these cornerstones, the possibility of using the DT as a way for building new

compelling applications will be played. The research activities are undergoing in many institutions and organizations. In this document, see next chapters, the focus is on a more convincing integration between descriptive and prescriptive modeling, chapter 4, supported by a general purpose sensing approach for collecting data (chapters 3 and 5). The DT framework chosen for development is FIWARE (RT.18) with extension for better support modeling and general purpose sensing.

⁶<https://www.ge.com/digital/applications/digital-twin>

GENERAL PURPOSE SENSING

*We are drowning in information but
starved for knowledge*

– John Naisbitt

3.1	General Purpose Sensing: Problem Statement	22
3.1.1	Definition of General Purpose and Synthetic Sensing . . .	23
3.1.2	Applying General Purpose Sensing to Large Environments	24
3.1.3	The Role of Data Fusion to Support Synthetic Sensing . .	25
3.2	Digital Twin and General Purpose Sensing	27
3.2.1	The Problem Statement	29
3.2.2	Needed Technologies and Functionalities	30

The deployment of an Internet of Things infrastructure comprises the installation of sensors, a number of gateways with local and long-range communications, and computing nodes. One or more long range communication networks can be used to transport data from local devices and gateways to cloud resources. Depending on goals and intended usage of the IoT infrastructure, these deployments can cover large areas and require a substantial number of devices and infrastructure. Consequently, interworking issues and management procedures are to be considered to guarantee that the platform is up and running and provides effective services. Having a large infrastructure with many types of sensors, results immediately in a management problem. The lessons learned from IoT deployments (e.g., in TIM in Venice) are that a large infrastructure should limit heterogeneity in terms of devices and promote a large reuse of data acquired in the field (e.g., noise detection for measurement of noise pollution, but also for security needs).

Customers perspectives are instructional: for example, small to medium cities are interested in an IoT infrastructure, but they have to justify the investment respect to citizens needs and city budget. The city of Issy-les-Moulineaux, close to Paris, France, is coping with an increasing level of traffic intensity. It is mainly due to the fact that there are many enterprises within the city limits and people from other locations are using vehicles to go to work. The citizens of Issy are not interested in deploying a large Traffic Intensity IoT network because many of them do not own a vehicle (they use public transportation), but even if they perceive traffic as a city problem, they would be more in favor of an investment on other issues of the city, for instance, pollution measuring and control. The current traffic solutions are based on generally available intensity data offered by Tom-Tom, but they are not necessarily reliable or precise.

On the other side, the analysis of a large city like Madrid with more than 5000 traffic intensity sensors and over 20 meteo and pollution stations scattered all over the city is informative. Its actual capabilities, potential, are supported by a well-structured management process (e.g., the curation of open data portal). The activities put in place demonstrate the needed effort to be effective. This infrastructure requires a precise planning, deployment and operation to keep up to an acceptable level of service. The large number of different sensors, the need for interoperability, the disparate types of protocols and systems to manage are issues to consider when dealing with large deployments of IoT. So far, the predominant approach has been to deploy vertical solutions and increase the number of devices and means making the infrastructure larger and more complex. Is there a need for having a sort of multi-sensing infrastructure capable of accommodate the needs of citizens but also to derive additional information about events that impact on the city life? In the next sections, a different approach will be proposed and explained.

3.1 General Purpose Sensing: Problem Statement

General Purpose Sensing is a technological proposition [6] aiming at using a small number of fixed sensing capabilities to measure a limited environment. The usage of Machine / Deep Learning technologies is instrumental to infer relevant information about events and environment's changes of status. This concept has been then extended to a system of virtual sensors capable (after training and learning) to provide synthetic sensing measurements of an environment [50]. The problem addressed by the combination of General-Purpose (RT.6) plus Synthetic Sensing (RT.7) is the ability to employ a sensory infrastructure made of sensors capable of measuring the environment and not strongly associated to a phenomenon

and be able to infer specific measures. Synthetic sensing, in particular, refers to the ability of inferring new information from raw data or by "fused", aggregated data. Issues related to this approach are several:

- What are the sensing capabilities from which to derive the most of information? What sensing capabilities can be considered acceptable and "general"?
- What kind of environments are the best candidates for general purpose sensing?
- What are the most promising techniques for supporting synthetic sensing? How these relate to general purpose sensing?
- What are synthetic sensing computational requirements? What existing datasets are available and promptly usable for training? Is human supervision required?
- What are the limitations of the proposed approach?
- What kind of training and supervision is needed to support synthetic sensing and what is the associated cost/engagement?
- What is an economical and viable architecture that can be used and can scale up?

Some of these issues will be further discussed in this chapter.

3.1.1 Definition of General Purpose and Synthetic Sensing

According to [6], *"general-purpose sensing, is the possibility of using a single, highly capable sensor to indirectly monitor a large context, without direct instrumentation of objects"*. In this definition, a single sensor is placed in the environment. Its data and measurements are used to produce information about what is happening around it. Actually, the initial environments chosen for experimentation were limited, e.g., a kitchen, a house, a small office. Another important aspect of the general-purpose sensing is that the sensor is not strongly coupled with a specific object, i.e., is not measuring distinct aspects of an exact object, but it is measuring actions' effects of objects on the nearby environment. An example can clarify this aspect: a specific sensor can be strictly associated to an object, so that when this object executes an action, the sensor will detect it. General purpose sensing operates in a different way: for example, a general-purpose sensor able to detect sound levels and also record the noises generated by objects is deployed in an environment (e.g., a kitchen). The opening and closing of a fridge's door generates noises that the general-purpose sensor can detect and these sounds, with the help of AI techniques, can be classified as actions operating on the environment (opening and closing of fridge's door). The same general purpose sensor can capture other noises (e.g., the kitchen door opened and closed, a coffee brewing machine, and others) and can collect this information. A level of synthetic sensing (after training) can classify a large part of sounds and noises in the kitchen and associate each of them to objects and actions performed in the room. In this way, the single sensor can be used to identify and classify different noises corresponding to events occurring in the "kitchen" environment and associate them to actions and states of many entities. A specific sensor, instead, can precisely detect data associated to the object it is measuring, e.g., detect the opening and closing of the fridge's door. The sensor strongly coupled with the fridge's door cannot be used to achieve more general goals. A general purpose sensing, GPS, sensor (and its related software supporting synthetic sensing) can be reused and also its knowledge could be improved to recognize additional events. GPS sensors are not to be necessarily associated to human perceivable sensing, for instance, each electric appliance has its own energy profile and the analysis of electrical consumption can be used to determine which appliance has been used and for how long. So the opening of fridge's

door can be associated to several phenomena: the noise of opening and closing a door, the light up of the fridge's bulb and, possibly, the starting of fridge's engine. This means that many different GPS devices and features can be used to detect events and generate their classification and relate actions and data to objects. For large and complex environments, different types of GPS devices could be used. The captured data can be used in isolation to derive information, but also "fused" to extract additional and possibly richer information. The association between a classified event and a GPS sensor strongly depends upon the application of Machine and Deep Learning algorithms. The GPS sensors will trigger and take a measure, they will record related data and will inject these measurements into an upper layer of AI functionalities that will identify, classify and determine the type of events, associate them to the environment and generate state information that fully characterize the results of event (e.g., "the fridge's door has been open now"). In this way, synthetic sensing can be defined as *the ability to transform measures, basic data into relevant information about events occurred in the observed environment*.

3.1.2 Applying General Purpose Sensing to Large Environments

According to [51], IoT systems are increasing in size and over 26% of them, already exceed 10.000 devices. The predicted growth of these large IoT systems is expected to further increase. In the near future, more of 50% of IoT systems will have medium to large size. Specialized sensing can have a considerable impact on deployment and operation. In smart cities, the usage of induction loop sensors provides very precise measurements of traffic intensity, but their usage will be limited to a single task. On the other side, some traffic intensity infrastructures are already using "Video Sensing" to count, classify vehicles and provide additional information (e.g., accidents). Even fiber optics networks are possibly used as sensing capabilities through the application of AI techniques [52]. These could be considered examples of general purpose sensing. The combination of GPS and synthetic sensing is a move from specialized and expensive "mono-sensing" infrastructures to lower cost GPS infrastructure supported by a large software capability providing advanced synthetic sensing features. This architecture exploits the edge-cloud continuum: general purpose sensing measures are transformed at the edge into information that is then forwarded to more capable nodes in the cloud for further analysis and provision of services. This is a **softwarization revolution** for large sensing infrastructures. The complexity is moved from hardware to software, in particular by means of a Synthetic Sensing segment. For some application domains this shift could open up new possibilities and enabling new opportunities and services.

To prove the GPS hypothesis in a large environment, Smart City domain was chosen with the ambition to verify the possibility to accurately determine the level of traffic in city areas by using environmental measures (e.g., pollution or noise). The idea was to associate a signature made out of pollutants' or sounds' levels to a quantity of vehicles in a specific environment. Obviously in a city, pollution and noise are not generated exclusively by vehicles. The location conditions and the pattern of involved vehicles can vary, make it difficult to have a precise mapping between signature and number of vehicles. Before further investigating this approach, data fusion experiments with different features (pollution, noise) associated to traffic were considered and carried out. This was methodologically needed to initially assess the approach, its feasibility, and relevance. Some of the experiments and results are reported in the following section.

3.1.3 The Role of Data Fusion to Support Synthetic Sensing

Data Fusion integrates multiple correlated sources of data into a larger data set and use it for inferring new information by applying statistical or AI techniques. A large number of methods can be used to exploit data fusion [53]. In the specific case of general purpose sensing, "sensor data fusion", e.g., [54], is relevant: it is the capability of merging different raw data generated by several sensors measuring disparate but somehow correlated aspects of the environment. To validate the possibility to apply GPS to a large and challenging environment, an initial study about its applicability on Madrid was conducted. City traffic prediction was the focus of experiments utilizing data fusion techniques (assimilated to general purpose sensing). The experiments were conducted in steps. At the beginning, only traffic intensity data were considered and used (a sort of baseline). Traffic intensity time series of a set of Madrid's sensors were considered and an LSTM Neural Network was built to predict the traffic intensity values of selected sensors. This process resulted in a good Mean Absolute Error value. In a second step, different activities were carried out: a set of pollutants more correlated to traffic were identified (see Fig.3.1a showing results for CO, NO, NO₂, NO_x) and this correlation studied and exploited to predict more accurate values of traffic intensity. This analysis showed that, in the initial phase of the day, alignment

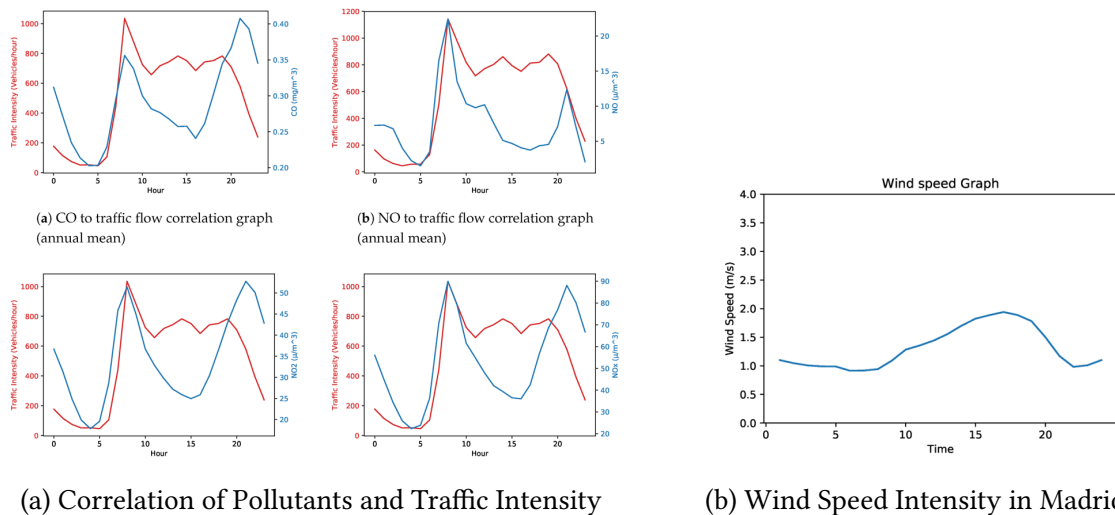


Figure 3.1: Correlation Analysis of Traffic Intensity, Pollutants and Meteo Factors

between traffic intensity and some pollutants' measures was good. In the second part of the day, this correlation was lost. After careful consideration, some meteorological factors were introduced because pollutants are reliant to wind and other meteo conditions. It came out that, all year round, in Madrid there is a light wind in the afternoon as represented in Fig.3.1b. This pushed to consider in the analysis and in the fusion data set a number of atmospheric data. Then, the issue was to identify the traffic intensity sensors "close" to pollution stations. They should properly collect data about more persistent and traffic related pollutants. This task, apparently easy, was complicated by two factors: the max acceptable distance between a traffic sensor and the closest pollution station, and the sensor's quality of data. Distance between sensors and pollution station was set to a maximum of 500 meters and the sensors were scrutinized for the quality of their data over time. A last issue was tackled: pollution data are generated every hour, while traffic intensity is measured every 15

minutes. The traffic intensity information was interpolated to the hour for using the same time frame. Data for three months were collected and curated for a few sensors and two months of data were used for training a LSTM and one for testing and validation. Fig.3.2a represents the Features chosen for the analysis. LSTM was chosen for its capability to deal

Feature	Value/Unit
Month	1-12
Day	1-28/29/30/31
Weekday	1-7
Hour	0-23
CO	mg/m ³
NO	µg/m ³
NO ₂	µg/m ³
NO _x	µg/m ³
O ₃	µg/m ³
Pressure	mb
Temperature	°C
Wind Direction	Angle
Wind Speed	m/s
Traffic Flow	Vehicles/h

- 3 Hidden layers (5,5,5)
- Dropout layer (0.7)
- Early Stopping (patience value 5)
- Look-Back Steps (168) representing one hour slot per day for the entire week
- Output layer (1)

(a) Selected Features for LSTM Definition.

(b) Hyper Parameters for the LSTM.

Figure 3.2: LSTM Features and Hyper Parameters for Traffic Pollution Prediction

with time series and the possibility to "remember" data along a week (there was a clear difference between traffic intensity during weekdays and weekends). The hyper parameters of the LSTM are represented in Fig.3.2b, some tune up was needed to avoid underfitting or overfitting. The achieved results are represented in Fig.3.3. The Mean Absolute Error, MAE, was used to evaluate the experiments' results. The combination of traffic intensity data with selected pollutants and meteorological features gave better results for each individual sensor taken into consideration compared to the pure traffic intensity prediction. More details and explanations about the approach and results are given in [29].

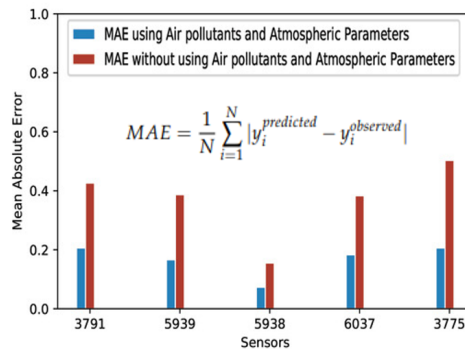
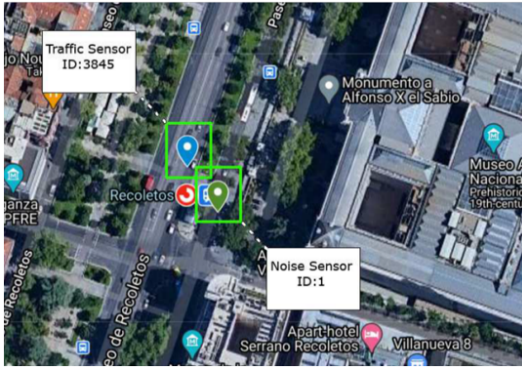
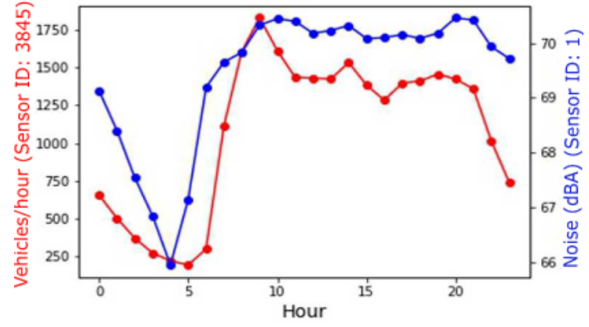


Figure 3.3: Comparison of MAE Values for Data Fusion Input and only Traffic Intensity.

Another set of experiments was organized: noise pollution data were fused with traffic data and used for traffic intensity prediction [55]. The experiments were based upon 3 months of noise data provided by the city of Madrid. Also in this case, traffic sensors taken into consideration were the closest ones to the location of noise sensors, as depicted in Fig.3.4a. These sensors' data measure how frequently a sound level is exceeding a fixed threshold of decibels, L_n measures how many times the level N of sound has been exceeded



(a) Location Proximity between Noise and Traffic Intensity Sensors.



(b) Analysis of Relationships between Traffic and Noise for Close Sensors.

Figure 3.4: Organization of Noise centered Experiments

over a determined period of time. L_{10} , L_{50} , and L_{90} are commonly used values. L_{10} is used for traffic noise assessment to perceive the continuous noise generated by flows of traffic. Before proceeding with the experiments, a correlation analysis was carried out to verify the relation between values of traffic intensity and noise. Fig.3.4b shows the curves of values under consideration. This correlation was deemed sufficient for starting a set of experiments. As already done for pollution data, 67% of data were used for training and 33% for validation. Features for this LSTM experiment were the noise levels and the traffic intensity (average per hour). The achieved results in terms of MAE are represented in Fig.3.5. Also in this case, the fusion of correlated data was providing better results than the single traffic intensity feature.

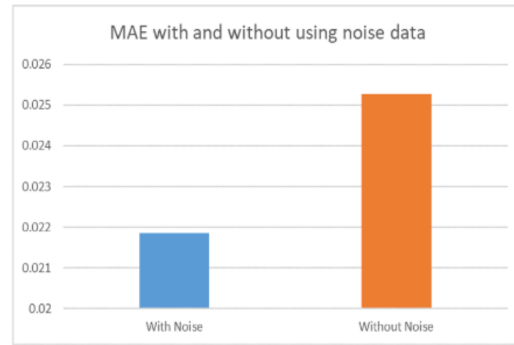


Figure 3.5: MAE for Noise - Traffic Analysis.

These results and the seminal paper [6] were instrumental to the attempt of generalizing the approach of general purpose sensing [7] to traffic intensity and to Smart Cities in general.

3.2 Digital Twin and General Purpose Sensing

Digital Twin and General Purpose Sensing are two concepts that, to the best of author's knowledge, were not deeply investigated together before the paper [7]. The proposal was to extend concepts presented in [6] to larger environments by using Digital Twin capabilities to represent the environment and objects operating in it (RT.5)). They could be identified by detecting some salient features by means of general purpose sensing and classified through synthetic sensing. This was instrumental to introduce the "Signatures of objects" concept. For instance, a diesel vehicle has a distinctive sound compared to a petrol or (even more) an electric car. This mark can also be so distinctive to differentiate the noise of diesel cars by brand or model. The signature represents a peculiar mark that can be used to

distinguish and identify a specific object respect to others. Shapes, noise, odors, or other marks can be considered and they can be identified by means of general purpose sensing.

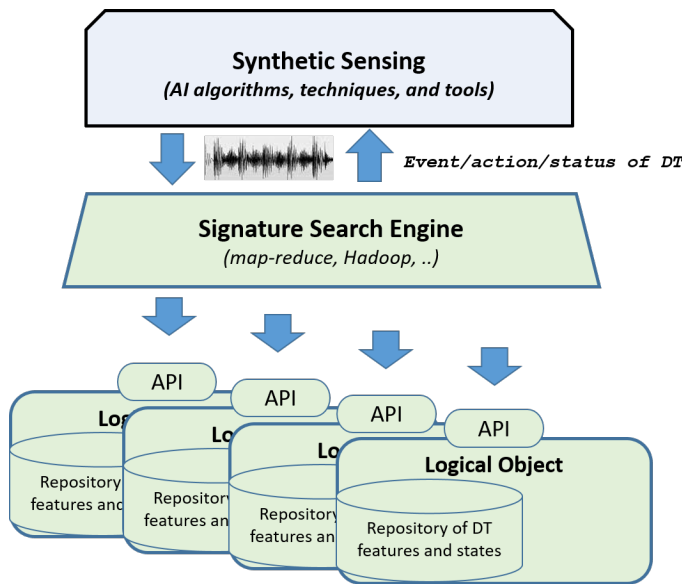


Fig.3.6 represents the case in which synthetic sensing functions fed by general purpose sensing can access a set of distinctive signatures to progressively identify, classify and recognize objects operating in the environment. The set of recognized features of an object can incrementally augment because for each of them specific crawling functions can be added. If a specific type of a vehicle has been identify, then aspects of it can be crawled finding additional information on the internet, e.g., pollution emission, noise, and more. These data can be used to create a

Figure 3.6: Signature as a Bridge for General Purpose, Synthetic Sensing and Digital Twin.

detailed view about the environment and objects. Another perspective is to focus on the incremental construction of a model representing the environment by discriminating "normal" behavior from anomalies. In the roundabout in Fig.3.7, two processes contribute to the environment representation: the identification of "normal" actions of objects operating on it (and the incremental acquisition of their features, such as usual directions, correlation between entrance and exit, no-go areas, as well as other parameters like average number of objects per hour, average speed, object type and the like) and the identification of anomalies. The normal situation "crawling" can also consider the specific objects passing by (most frequent type of object, then its brand, a few characteristics having impact on the environment, and so on) and creating an increasing environment model and its variations over time. The anomaly detection feature will progressively be more accurate when anomalies are spotted and understood. For instance, if two objects, vehicles C and D, are stopped in an area that is not usually occupied, then that occurrence could generate an alarm. As well if an object, like E, is in a no-go area, another possible alarm can trigger. Progressively

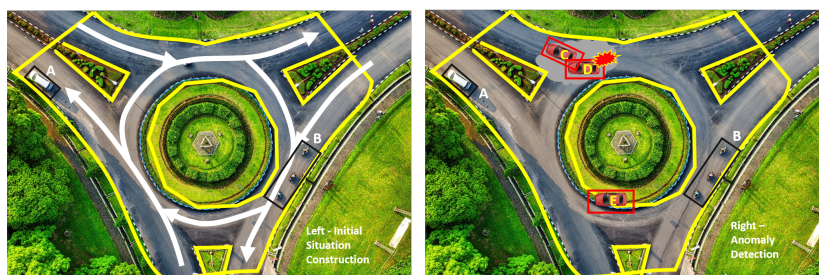


Figure 3.7: From "Normal Situation" Construction to Anomaly Detection.

an accurate bottom-up model of the environment and operating objects can be built. The bottom-up approach (RT.4) is a novelty: only a very basic environment and constituents model is predefined, and it is up to the DT software to incrementally build a factual environment representation and understanding of context (i.e., the status of environment and its components at a precise moment) and current situation (the warnings and anomalies detected). To speed up the process and to focus on specific features, some plug and play specialized software could be added to grasp the salient data and create the corresponding modeling. Fig.3.8 represents a software architecture for the bottom up construction of a

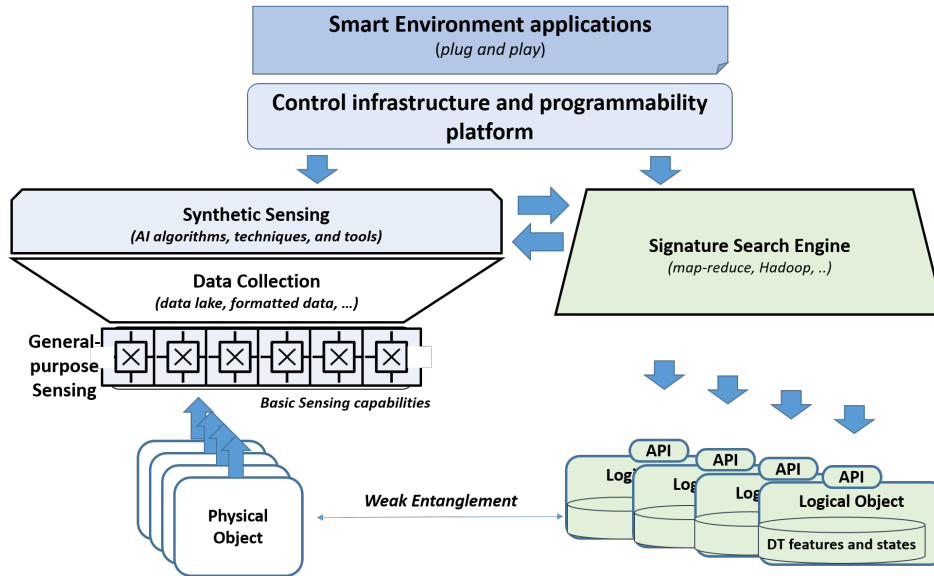


Figure 3.8: An architectural View for Bottom Up Construction of Digital Twin.

Digital Twin model. Its characteristics are the usage of general purpose / synthetic sensing capabilities to identify signatures and to incrementally extend the mapping between objects and the environment, and a set of extensible programming capabilities that can support the specialization and a more purposed modeling of the environment.

3.2.1 The Problem Statement

The approach described in section 3.2 can be considered a sort of research road-map to be pursued in the validation of the proposition of combining the DT and general purpose sensing. There are two major assumptions to be clarified and proved:

1. The applicability of general purpose and synthetic sensing to the needs of an effective Digital Twin representation.
2. The verification of effectiveness of a bottom up approach in building a Digital Twin.

They are real challenges that may open up to new DT applications. However, even if they prove feasible, there is a threat to cope with: the approach practicability. General purpose sensing moves costs and issues from a specialized hardware infrastructure to a sophisticated software architecture transforming raw data in valuable information by means of data fusion and other transformations. If it is so sophisticated to require a huge software infrastructure or if the accuracy or availability of transformed information is not compatible with application requirements, the entire effort could be very academic, but useless at an industrial level. Determine usability and limits of the effective behavior of a GPS fed

Digital Twin is important. A bottom up approach can exacerbate these problems, because it introduces a need to incrementally build a viable model of the environment and its operating objects. This exceeds the typical time for training of Neural Networks, and requires a continuous tune-up, improvement of algorithms and DT logic. Even if plug and play functions can help to optimize the capture of relevant features, the creation of a stable and effective bottom up model can require time. Initially, it is difficult to differentiate between normal behavior and anomalies. So that some human supervision could be useful to cut training time. In any case, the challenge posed by General Purpose Sensing Digital Twin is worth to be considered and finding its limitations is a relevant research topic.

3.2.2 Needed Technologies and Functionalities

Despite the current efforts towards definition and usage of Digital Twins, a classification of its types is not generally available or well consolidated. The approach proposed in [39] differentiates DTs on the basis of their behavior, usage of AI, and degrees of freedom, from passive to autonomic digital twin. The "*bottom up Digital Twin*" is a basic autonomic DT capable of increasing its environment representation along time by progressively adding objects and differentiate situations. The current trend in development of DTs is to create a top-down representation by designing an accurate prescriptive model during the specification phase. For a bottom - up DT the design is different and an effort in identifying technologies and tools for progressively increase the prescriptive model accuracy at run time is a research goal. Another aspect related to this investigation is the identification and composition of different objects' behavior (RT.12) (represented themselves as smaller DTs) within a domain (represented as a large DT) [56]. Upfront to a large set of usable technologies, a major critical point is the identification, combination and integration of them for reaching DT representation goals. This problem is essentially unexplored with respect to the bottom up approach. The starting basis for identifying technologies and tools supporting this type of DT is the vast land of Artificial Intelligence techniques. Good results are expected from technologies for image recognition, transformers and knowledge graphs. However, the initial steps will consider a limited environment with a specific set of problems to handle and possibly govern (e.g., traffic "understanding" on a specific crossroad). So, a restricted set of AI technologies can fit the goal. The proposed method is to start with a bottom up approach for a well defined problem and to use a signature based framework for measuring, identifying and classifying actions and agents of the environment. If results are accurate, to enlarge the investigation to specific features of the environment. In perspective, the idea is to establish a set of signature technologies that work well together and can be used to collect, aggregate and reuse data to improve the representation of environment and its behavior. For data management, existing ontologies and data models will be the starting point, but generalizations of them (or fusion of different ones) will also be possible to increase the capability to represent complex objects and not well defined situations. Also, technologies related to reasoning and situation awareness will be explored and possibly adapted to contribute to the construction of a effective solution. Existing software infrastructures, e.g., Eclipse Ditto or FIWARE, will be used to exploit the rich set of functionalities supporting the programmability and management of large quantity of data.

Part II

Contributions

DIGITAL TWIN EVOLUTION

Fortunately, most human behavior is learned observationally through modeling from others

– Albert Bandura

4.1	Architectural Framework	34
4.2	Modeling Problem	36
4.2.1	Descriptive Modeling	36
4.2.2	Prescriptive Modeling	40
4.2.3	Multi-Facet Digital Twin Problem	41
4.3	How Modeling Fits Into Architectural Framework	41

This chapter provides a description of DT research challenges with actual and future contributions expected from on-going activities. The research goal is to contribute to the definition and construction of an effective architecture that helps developers and practitioners in implementing well-formed Digital Twins. The contributions aim at easing DT modeling of prescriptive parts and extension of descriptive capabilities to encompass different perspectives and views on aggregated physical objects. Additional research objectives are pursued to overcome the complexity of this approach and its wide applicability.

The architecture's definition is based, as a starting point, on the adoption of a well known basic DT framework and its progressive extension to encompass functional segments that augment its "expressive power" (RT19). This architectural work has resulted in the identification of needs and drawbacks of current major efforts in DT platform definition. FIWARE platform has been chosen because of its extensive usage within European and National projects (RT.18). Due to the complexity in designing a Digital Twin, the availability of templates with most common "behaviors" of different classes of logical objects would be an advancement with important consequences (RT.16). The behavioral modeling is also important to support a characterizing property of the DT: simulation and its proper reuse of a prescriptive model (ref RT.11). The programmability of the platform should also guarantee DT composability, (RT.12). This problem comes with two challenges: the integration of different prescriptive models of the same DT (ref RT14); and the support of many facets of an object by means of different but interoperable descriptive models (RT.15).

This chapter will explore similar issues that can drive the evolution of DT frameworks.

4.1 Architectural Framework

The landscape of Digital Twin reference software architectures is fragmented, there are standards, e.g., [57] and proprietary solutions (e.g., Microsoft [23]), open source ones (e.g., FIWARE [22] or Eclipse [33]), novel approaches (e.g., [58], [59], [60]) or specific ones for application domains (e.g., [61]). There is not yet a convergence towards an agreed reference framework. In this situation, a FIWARE solution [22] is taken as a reference within the context of French National Project C2JN ¹. The proposition of the project is to integrate and/or develop tools and solutions for building a software platform capable of supporting the development of DT based solutions for different problem domains. It is also designed to operate on edge-cloud continuum. C2JN platform exploits existing tools and mechanisms, while it only extends or requires development of missing features. The central element is an infrastructure of interworking FIWARE Context Brokers [22]. Data flows are coordinated by a master Context Broker. The main sources of data are: batch data or IoT and Edge generated flows (typically real-time). They are treated as raw data and undergo through curation and injection to ensure they are formatted, usable and valid before being stored in the Context Broker (it can be seen as an implementation supporting the memorization property). An example of this injection chain (RT.13) is provided in [15]. Fig.4.1 represents functionalities types that support the platform's core goals:

- Edge - Cloud Continuum infrastructure. It is an Edge subsystem comprising Edge devices, and a Cloud subsystem comprising nodes for heavy computations. Edge and Cloud resources are managed by means of existing Orchestrators (e.g., K8s or K3s [62]). The edge comprises Jetson and RaspberryPi devices, and this requires inter-

¹<https://insight-signals.com/insight-signals-the-c2jn-project-round-table-cloud-continuum-and-digital-twins/>

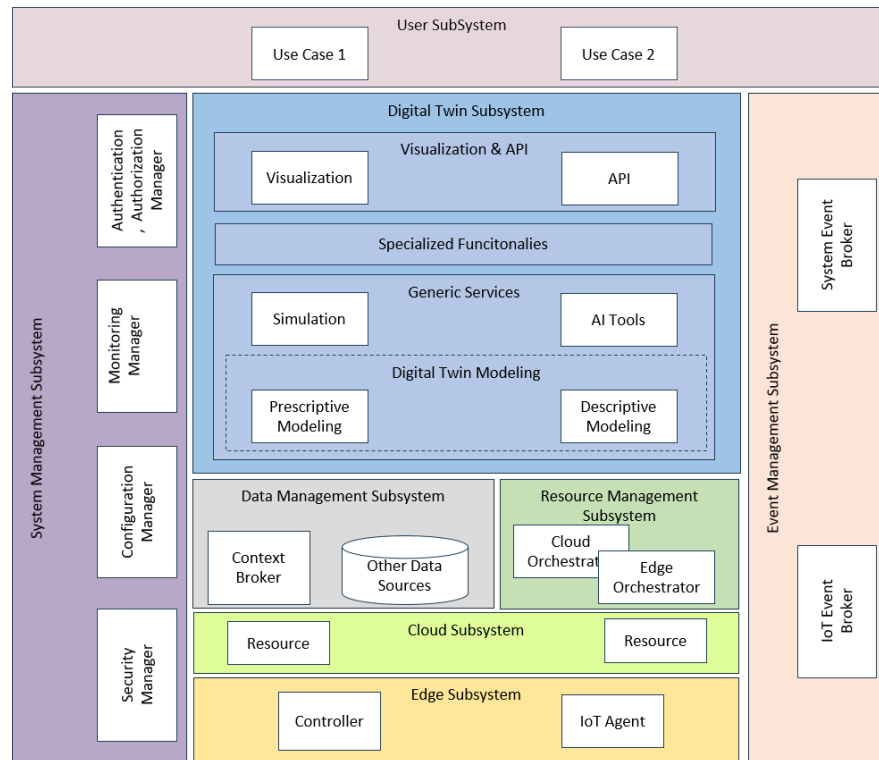


Figure 4.1: A Reference Architecture for DT Applications based on FIWARE.

- working between orchestrators comprised in the Resource Management Subsystem.
- Event Management Subsystem is made out of mechanisms and tools supporting the passing of events between different components of the platform (if and when needed). They are instrumental to support the work of FIWARE Context Brokers.
- The System Management Subsystem is offering functionalities for guaranteeing the platform security and privacy of data and activities.
- The Data Management Subsystem comprises tools and processes needed to collect data from Edge and other external sources, transform them using well defined Data Models and inject them into the set of Context Brokers. They are the source of data that DT functionalities use to achieve their goals.
- The Digital Twin Subsystem provides functions, tools and mechanisms to drive the execution of DTs and supports user applications needs, in particular: prescriptive and descriptive models interplay, simulation and prediction integration, special functions for supporting the DT (e.g., plug and play ones for supporting bottom-up modeling), visualization, and APIs for augmentation and access to DT services.
- The User Subsystem comprises applications, services, and functions that make use of the DT representation in the user domain.

C2JN project focuses on the definition of a viable Digital Twin framework capable of supporting different use cases. The selected ones are: a Smart City application for traffic monitoring, prediction, and simulation; an Edge-Cloud Continuum monitoring system by means of a Network Digital Twin (RT.20) representing different features of the infrastructure including measurements of CO₂ footprint of its components[63] (RT.21). An important aspect of the architecture is that the Edge cloud management infrastructure is represented as a

Network DT itself, i.e., a sort of recursive management is implemented. This architecture is experimental in nature and is used to verify working and research hypothesis, and as a testing lab for DT development in a distributed infrastructure.

A study about existing frameworks and the level of supported modeling has been carried out [49] and it is continuously updated to keep track of improvements and progress in this sector. If important contributions will emerge, they will be readily integrated in the Service Architecture Lab of TSP to experiment and progress on this subject matter.

4.2 Modeling Problem

The Digital Twin is a contextualized model representing states and actions of a system (RT.3). In the industry, best practices for descriptive and prescriptive modeling is consolidated [13]. However, the interpretation of these modeling requirements and applicable techniques are divergent. Modeling is a major concern for the success of DT implementation. For more information and understanding of this issue and how to tackle with it, see A.2. Aiming at a fully fledged Digital Twin, the reference architecture for C2JN and this research have been designed to progressively accommodate for supporting interoperability of modeling options. Some results are presented in the following subsections.

4.2.1 Descriptive Modeling

Activities related to Internet of Things and the interest for integration of data into a framework usable for Digital Twin representations led to a strong attention towards DT supporting platforms. The viable possibilities were two: the Eclipse Ditto and the FIWARE Context Broker. Both were explored in research activities, e.g., Eclipse Ditto has been used in [64], [65], [66] and [67]; and FIWARE has been exploited in C2JN and another European Project, SALTED [68]. The SALTED experience has highlighted the importance and need for curation of data and their organization into Data Models or ontologies. The careful definition of properties and relationships between data are instrumental to a powerful DT descriptive model. In addition, the ability to apply knowledge graphs or other cognitive techniques to several data sets formatted according to NGSI-LD [69] (or other languages) increases the data value and the "information density" (i.e., the ability to extract, infer or create new data) of available datasets (RT.2). This is extremely important for the DT descriptive framework. These characteristics have been discussed, experimented, and assessed in the SALTED Project and reported in [15]. The best practices and experience as well as available tools (the Context Broker and some data conversion tools) derived from the development and usage of a powerful injection chain helped in data fusion experiments and in development of specific Data Models serving as the basis for a descriptive framework of Digital Twins according to NGSI-LD[69] [22]. This led, for instance, to basic analysis of Madrid traffic and pollution data [68]. This was instrumental to understand how large amounts of data can be used and put together to determine the behavior of a large and complex system as a city. These experiences found also a re-use in C2JN project's activities. The FIWARE Smart City data models were reused, improved and adapted to the City of Issy for predictive purposes [70]. In the context of Edge-Cloud Continuum, it led to the definition of an extended data model supporting the function and needs of a Network Digital Twin [63]. These efforts are briefly described in the following sections.

Data Modeling for Smart Cities

The adoption of a FIWARE based architecture and the participation to FIWARE oriented projects have pushed towards the exploitation and extension of already available FIWARE Data Models. The Madrid Open Data site offers a wealth of data about many aspects of this city and it has a large and well curated archive of traffic data². The city of Dublin also makes available traffic data for flows within the Dublin county³. Cities variously store data as shown in Fig.4.2 for traffic related data. The first issue is to make the data comparable

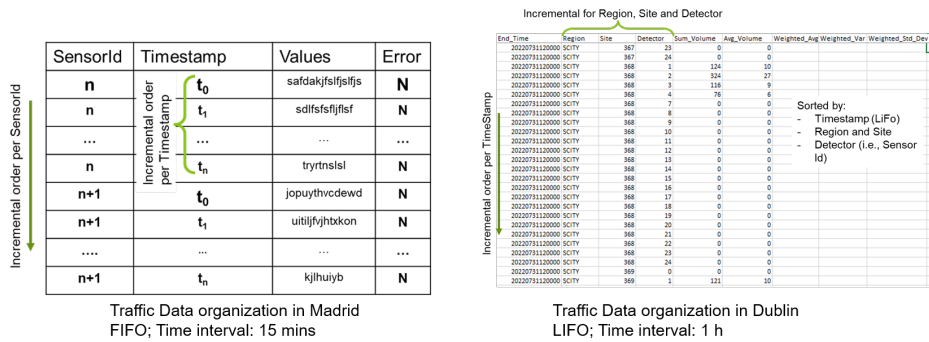


Figure 4.2: Madrid and Dublin Data Structures for Traffic Intensity

and to transform them into an interoperable format. Similar issues are also encountered for pollution data. The chosen data models were those of FIWARE⁴ for traffic, TrafficFlowObserved⁵, and pollution, AirQualityObserved⁶. Data compatibility issues were several and required many activities to achieve interoperability and alignment to chosen data models. The processes for feeding the data models are described in [71]. The major issues to cope with are:

- Missing data and admissible padding by normalization.
- Normalization to time periods (e.g., Madrid every 15 min, Dublin every 60 min).
- Identification of errors and actions to take.
- Replication of data and how to cope with them.
- How well the available raw data fit with the model (too many missing data).

The Smart Data Models are designed for completeness of data, in fact they comprise a large description and definition of data. Mapping the existing data on these models can lead to a lack of data that for most complex application can be a problem. However, applications can be designed in such a way to use the "slim" information. In this case, they will miss a bit of data model expressive power, but information will be usable as much as the original data sets, though with a higher level of interoperability and further "padding" possible. A practical example of injection chain usage for populating data models and the possibility to easily fuse the data is provided in [5]. Here, the injection chain and the Context Broker capabilities are used to collect, curate and aggregate traffic, pollution and meteorological data in different time periods and normalize them. Data referring to typologies of urban

²Data can be collected by surfing to <https://datos.madrid.es/portal/site/egob> and then searching in the catalog for "Tráfico. Datos del tráfico en tiempo real"

³e.g., <https://data.smartdublin.ie/dataset/dcc-scats-detector-volume-jan-jun-2023>

⁴<https://www.fiware.org/smart-data-models/>

⁵<https://tinyurl.com/yc39m58b>

⁶<https://tinyurl.com/msjatmka>

areas of Madrid (Urban-North, Urban-South and Suburban) are compared in terms of pollution, traffic intensity before, during and after the COVID. Impact of meteorological data and difference in weather condition over the years are also considered to make the comparison fair and methodological correct. This activity could be a basis for creating a passive Digital Twin capable of collecting and organizing these data to represent the variation and similarities between different area of the same or several cities (data from Dublin, Barcelona and Santander were also considered during the experiments).

Another example of extensive usage of data models integrated with predictive and simulation is in the paper [72]: in order to derive traffic intensity data, Google traffic distance matrix⁷ information is collected for a set of source - destination couples covering a certain area (Issy-le-Moulineaux) and transformed into a FIWARE TrafficFlowObserved data model. These data are fused with RATP Bus real-time position⁸ (for exploiting the relation between traffic intensity, schedule real time of buses passage in a same area) and other meteorological data that can have an impact on traffic intensity.

These experiences and experiments demonstrated that the existing descriptive modeling capabilities (essentially based on FIWARE but also Eclipse and DT Definition Language as defined by Microsoft) are capable of representing data for the usage of Smart City DT.

Data Modeling for Network Digital Twin in the Edge-Cloud Continuum

A research topic (RT.20) related to DT descriptive modeling is the Network Digital Twin. There is an increasing interest in Network Digital Twin [73][74]. The current definitions address the representation of a complex network - computing infrastructure by means of a DT. Advantages of a NDT are several such as be able to use historical data for prediction, simulation for optimization of resources, behavioral model for exerting control on the entire infrastructure, and data models for supporting interoperability and providing granular/abstraction capabilities over parts of the infrastructure. NDT representation is proposed mainly by Communication Service Provider, CSP, or telco Vendors. Consequently, it focuses on a Network view fully describing the nodes and the equipement of the Operators. An important part of the computing infrastructure is under-represented (e.g., geographically dispersed but connected Service Providers' Data Centers). Actually, the view of a large edge - cloud continuum should comprise networks, strongly tied with a computing infrastructure . A NDT represents a large distributed system segmented into different cooperating and communicating administrative domains connected to final users. Hence, Providers and Customers perspectives should be represented to fully grasp the complexity of this infrastructure. Needs and requirements of applications and management systems of different stakeholders are to be considered. The NDT cannot be only focusing on one stakeholder, the CSP. It is necessary to extend it for representing:

- The segmentation of different domains connected by means of specialized networks. Segments could be Edge, Core, Cloud connected by access core and providers networks (as the current CSPs trend proposes). They can be generalized into Network or, even better, Infrastructure Segments that may have varying capabilities and different response times depending on geographical proximity to certain customers.
- A model of an edge infrastructure made out of smaller or less powerful computing resources respect to gigantic Cloud's Data Centers and how they relate to each other

⁷<https://developers.google.com/maps/documentation/distance-matrix/get-api-key>

⁸<https://prim.iledefrance-mobilites.fr>

(hierarchical or fully interconnected infrastructure).

- Abstract views of the infrastructure from application developers, service providers, and all the stakeholder that may have the need to have a holistic view on the distributed and segmented infrastructure they use.

To introduce these facets, a Network Digital Twin ontology⁹ subsequently transformed into a FIWARE Data Model¹⁰ have been designed. In paper [75], an initial cloud based ontology is presented. It was taken as a basis for developing a larger Edge-Cloud Data Model described in [63]. The specification is quite complex and still in a consolidation phase, but under development and testing within C2JN project. Fig.4.3a represents a "segment" within

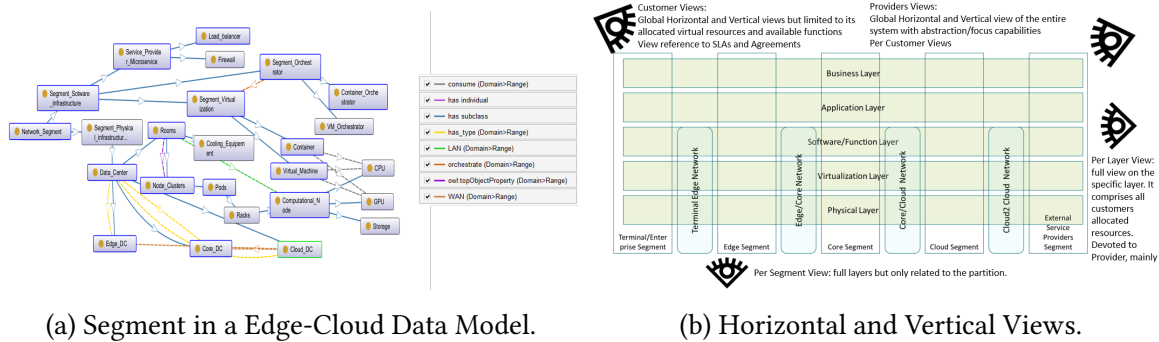


Figure 4.3: Network Digital Twin Representation and Support for Stakeholders.

an Edge-Cloud Continuum system, i.e., an aggregation of resources co-located in a specific geographical area. Proximity with customers is a property of this segment (e.g., Mobile Edge, or Edge as defined in amazon cloud infrastructure). This characteristics ensure a low response time if (and only if) the large majority of components of the application are in this segment. A latency and cost analysis of edge capabilities is discussed in [76]. Fig.4.3b represents the target views offered to different stakeholders by the Data Model. Two main views are generally possible: an horizontal one in which different level of functionalities are represented; and a vertical one, on which different computing and network segment are represented. The Physical layer represents the available physical resources (from servers to switches to ports and the like); Virtualization layer represents the virtual resources made available and those used by the customers; Software/Function layer represents the used infrastructure features offered and used by customers (e.g., a virtual network function, VNF); Application layer represents the different components, microservices comprising the applications; Business layer represents views and requirements (like SLAs) as requested by the customers. In this way, queries about all the info related to a layer can be posed: what are the Virtual layer resources that a customer is using all over the infrastructure?

Vertical segments represent the same layering information but referred to a functional domain (a segment) of the entire infrastructure. Here the model can answer to queries about the usage of resources within a domain such as "what are the resources of the segment that a Customer (or a provider) is using"?

On top of these representations, a NDT can apply optimization policies for improving the working of a layer across the entire infrastructure, a segment and the usage of resources

⁹originally developed with Protege tool available at <https://protege.stanford.edu>

¹⁰still under consolidation, available at <https://github.com/SyedMohsan/>

from the customer perspective. An implementation of this logical structure for a NDT to corroborate the definition is undergoing. In a edge - cloud implementation in TSP lab, the Prometheus tool¹¹ is used to scrap data from physical and virtual resources. This information is formatted into NGSI-LD and injected into a Context Broker accordingly to the described Edge-Cloud data model. Different logic and applications are under development to support relevant customers' and vertical views on segments of the infrastructure. Practical results will be instrumental to consolidate and improve the Data Model specification.

Descriptive Modeling: necessary but not sufficient for a DT Implementation

Activities on DT descriptive modeling are the basis for creating a data infrastructure to feed prescriptive models representing the system behavior or at least AI software that predict future states. Without a behavioral representation, the DT is "passive", i.e., it can merely report system change of status by means of well formatted data (passive DT). The other possibility, as depicted in Fig.A.2a, is to use the collected historical data for predicting future system changes of states (a predictive DT). Indeed, this covers a large part of actual implementations that, on top of predictive ML / DL functions, add a layer of rules like "if <this> then <that>". This is an acceptable implementation, but it does not exploit the capacities of a fully fledged DT. Descriptive modeling is the most developed and stable part of DT modeling. However there are still things to define and research, for instance more efficient and comprehensive models for specific problem domains and a general way for supporting the multi-facet description of a system into an integrated view.

4.2.2 Prescriptive Modeling

The prescriptive or behavioral modeling¹² of a DT is mainly a peculiar and vertical specification. Little generality is sought for and specific solutions are the norm. This modeling is supported by methodologies that are especially applied in a top down fashion. MBSE [16], [25] can be used, and, typically, the implemented solutions are not reusable and shareable. This is due to the intrinsic issues of modeling a specialized system and to the fact that providing a working model of a proprietary system to competitors could be a business mistake. In this way, solutions are proprietary, very segmented, and specific. In addition, with these methodologies, the representation as a DT of already existing systems is very difficult and left to the developers also when some re-use of models could be useful.

Looking for Prescriptive Templates

The contributions [49] and [77] have identified a possible research path in the definition and reuse of prescriptive templates for an initial small set of typical behavioral models of a system or reusable components. These templates could be offered to programmers to start with a well-formed skeleton of actions and expected results representing the intended conduct of the DT or component and the relations with their counter-part system. The working hypothesis is to design effective templates that comprise for each DT and its subsystem the following elements:

- Goals and subgoals of the specific entity represented in the DT model.
- Constraints and Rules that the entity should always satisfy.
- Basic actions that the entity executes and performs. These actions represent physical actions or calculations performed to better align the behavior to a mathematical /

¹¹<https://prometheus.io>

¹²so far they are synonyms, in 6.1 they will be better substantiated highlighting differences.

- physical model. A set of actions are a Task aiming at fulfilling a (sub)goal.
- Messages and interaction means between the system entities of the model to exchange data, information and also tasks, and actions (for prioritizing).
 - Resources and/or means to be allocated, used, and released for the execution of tasks.
 - Locations and places where actions, resources, or constraints are located.

These elements can be related and structured according to a simple "programming" language that will support controlling primitives (like if-then, function calls, and management of events). From a modeling perspective different templates should be considered and designed, formalized. Some of them should use rules, constraints and a set of basic actions to describe only the system and its goals. In these cases, systems are insignificantly or lightly impacted by the environment (e.g., a oil vessel used to distill oil). For other systems and related templates, it is important to determine the environment impact on the system behavior. Other templates could be developed considering the DT goal, for instance in the case of a Network DT, its goal is to monitor resources and to actuate optimization policies. This can result in a set of templates to be specialized to different environments or goals. The goal of this research activity is to identify and build different modeling structures of possible DTs as templates and to provide them in terms of primitives pertaining to a set of characterizing behavioral elements, supported and defined by means of MBSE methodology and tools. Since this research has not reached maturity and not relevant articles have been published, A.3 is providing an extensive description of intended templates.

Scenario Based DT

Other Digital Twins may be hypothetical, in the sense that they are representations of systems that have still to be built and they are used to determine by simulation and modeling relevant features that the real system should have. They will be further discussed in A.3.4. This type of DT is extremely promising because it can help in planning complex activities or processes in smart cities [78]. The scenario based DT can be used to optimize the available resources or to determine what resources are needed to reach the expected goals (e.g., as in the case of a recycling process discussed in A.3.4)

4.2.3 Multi-Facet Digital Twin Problem

Digital Twin concept was applied to Cultural Heritage [30], [31] and to the representation of different aspects of the same object (RT.22). With the increase of usage of DT based solutions, the need to dynamically decompose and re-compose different aspects, facets of components will be a strong requirement for an effective representation of complex objects. Also this topic is in its initial stage, A.3.5 provides the basis for a research path.

4.3 How Modeling Fits Into Architectural Framework

The discussed modeling features and capabilities are not available in current systems. Prominent companies have developed entire systems for supporting their needs in this sector. There is a wealth of knowledge, developments, and Intellectual Property Rights behind these solutions. They represent competitive advantages, that very unlikely will be fully disclosed for the sake of creating a fully fledged open source system supporting all the discussed aspects. On the other side, some tools like Open Modelica [79] offer a few of these capabilities, but they do not integrate all the needed modeling tasks. Ideally there is a research space for the implementation of an Integrated Development Environment devoted to DT construction (RT.23).

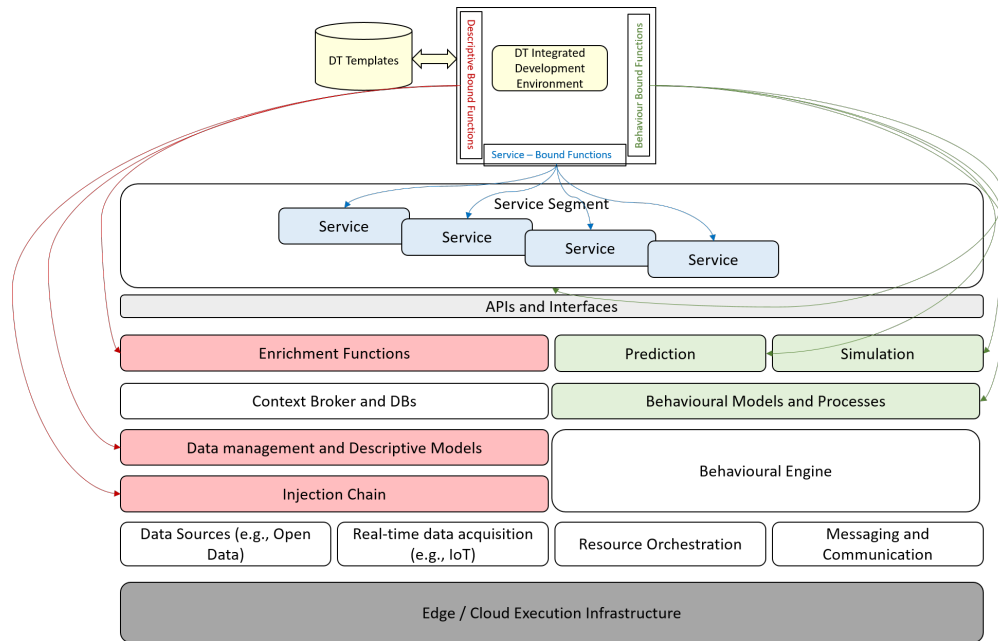


Figure 4.4: An integrated IDE for DT Implementation

This IDE, represented in Fig.4.4 has to support the creation and development of DT based applications and services (the light blue parts) as well as the improvement and modification of descriptive modeling (the red part) and the prescriptive components (the green part) in a seamless way. The deployment, execution and management parts should be managed by other tools and mechanisms. The IDE should support templates definitions, reuse (i.e., a repository of well defined and proved skeletons of DT behaviors and data ontologies), an easy integration with methodology for MBSE, and tailoring of existing components. The availability of a general purpose control language, the introduction of mathematical and physical models within this language, tools and means guaranteeing the ability to use multiphysics would be some of the required capabilities of this DT-IDE. In addition, it should have the ability to represent the needed steps for negotiation between interacting entities within different environments and under specific conditions. A fundamental aspect of the IDE is the ability to represent the Environment (the context) and to highlight the current situation to allow an increasing awareness of the status of the physical system. The complexity of this IDE is great, but it is a needed tool for enabling a larger usage of DT solutions also to enterprises and practitioners that do not have a consolidated experience in the implementation of Digital Twins.

GENERAL PURPOSE SENSING: THE WAY FORWARD

Google Earth is an incredible resource because from hundreds of miles in space, we can zoom in, and we can find things. Everyone always looks for their house first. That is the tip of the iceberg with remote sensing.

– Sarah Parcak

5.1	General Purpose Sensing at work	44
5.2	Generating Data at the Edge	45
5.2.1	Opportunistic Digital Twin	46
5.2.2	Looking for Bottom Up Models	47
5.3	Reusing Data in a Smart City	48
5.3.1	Traffic Intensity and Pollution	48
5.3.2	Traffic Intensity from Google Data	49
5.3.3	From Redundancy to Optimization	50
5.4	An Architecture for General Purpose Sensing	52

5.1 General Purpose Sensing at work

Usage of general purpose sensing, GPS, in large environments and the relationship to signature mapping has been described in [80]. This working hypothesis (RT.5) needed to be proved in the field. Some experiments, like those on traffic and pollution [29] or noise [55], were giving promising results, but the data capture was specialized and ensured by well consolidated procedures put in place by large organizations. This was, somehow, guaranteeing good quality data. Considering the potential needs of data in small medium size cities, experiments in the collection of GPS data were carried out aiming at assessing the approach viability. The initial idea was to challenge GPS hypothesis by using a minimal set of sensors over an open hardware framework. Video was excluded because of privacy concerns. However, it proved to be one of the richest sensing means. The essential steps for general purpose sensing exploitation are: i) collection of general purpose data (e.g., noise or pollution); ii) transforming them into usable datasets; iii) training of algorithms and ML / DL tools to recognize and transform these data into wanted information (synthetic sensing, e.g., data intensity); iv) training ML / DL algorithms to predict information (e.g., traffic intensity); v) ability to visualize, possibly through a Digital Twin, the acquired data; vi) ability to provide a set of APIs and programmable functionalities that can support the construction of services. Initial experimental setup was using noise sensing for "measuring" Traffic Intensity, TI. This was instrumental to constructing a general purpose sensing infrastructure and deriving a viable architecture supporting the previously described functionalities. Two technical challenges were considered: a general purpose sensing (open) hardware and its internal software structure; a higher level software organization and the ability to deal with data. This comprises AI tools for analyzing and transforming the data acquired in the field. The practice led to the setup of a simple hardware infrastructure based on two device types: Resource-Limited Device (RLD), e.g., RaspberryPi; and Powerful Edge Device (PED), e.g., Jetson. They are instrumented with low-cost sensors for detecting noise, pollution, temperature and others. During the operation, a video camera was added.

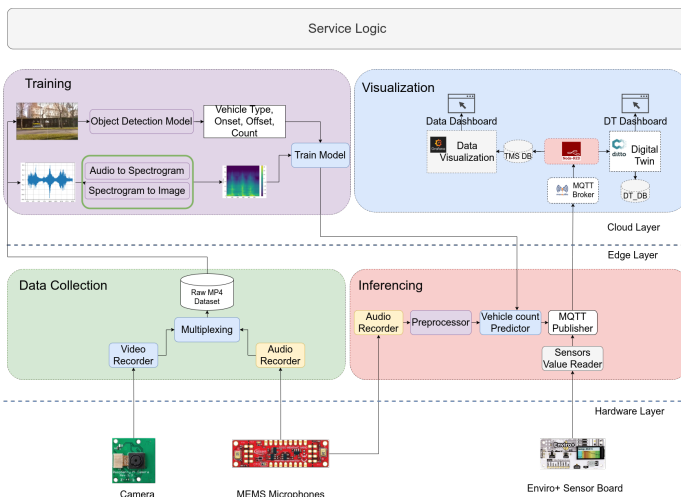


Figure 5.1: General Purpose Sensing System for Traffic Intensity Detection from Noise Signatures.

An issue during experiments was the lack of suitable noise datasets for vehicle recognition. The introduction of a video camera was instrumental

As represented in Fig.5.1, the system comprises four general functions: Data Collection, for gathering environmental data; Inferencing, a general purpose sensing inference engine; Training, an environment for training and assessing algorithms; Visualization, a service logic with visualization, a DT representation of the environment, and an API front-end for programming. The system architecture was the result of different studies [64], [65], [66], [67].

for collecting road footage and to use it with YOLO for vehicle detection and counting [81]. The same general purpose sensing devices were used and a dataset associating vehicle and noise in a specific area was created by YOLO and used for training the inferencing algorithm. Similar mechanisms were adopted to collect data for validation. As described in [67], Video Transformers, ViT, were used to classify the vehicles and noise. This information is transformed into images and compared with baselines to determine similarities. This can be considered as a sort of video signature analysis. Light-weight ViT-based models, such as TinyViT [82], EdgeViT [83], and MobileViT [84] were tested for audio classification and events detection as well as other models, such as MobileNetV3 [85] and ResNet [86]. More detailed descriptions of experiments setup, implementation and their results are reported in a Master Thesis [66] and an article [67]. Experiments were satisfactory (especially after the video introduction) and they demonstrated that under certain circumstances, the methods are valid. Some highlights are:

- The quality of used data sets, obviously, influenced the results.
- The system is easily reproducible and these mechanisms provide a relatively simple way to create mobile or temporary monitoring points for traffic.
- The DT (a passive one for the moment) has the main function of collecting data and organize them.
- The importance of video as a fundamental sensing means clearly emerged. From video, many different data can be inferred and transformed into information (more in section 5.2.2).
- The signature approach is practicable and may be aligned with the increasing adoption of Transformer based technology [87]. This is a possible research path (RT.5) to be further investigated.

5.2 Generating Data at the Edge

The basic concepts for developing a GPS architecture integrating the DT and signature were introduced in [7] and are discussed in a Master's Thesis [64]. The starting point was the signature framework represented in Fig.3.8. General Purpose Sensing infrastructure is capturing signals and measurements related to entities in an environment. After training AI tools, objects are identified and classified according to the way they "modify" the environment and a signature is determined. They are collected and incrementally "stored and matched" with known and existing ones. The sound of a vehicle is a distinctive feature of a product (think about a Harley Davidson motorcycle or a Porsche car). Even electric cars manufacturers are working on distinctive sounds (i.e., signatures) for their vehicles [88], [89]. The collection and organization of signatures in a Repository could be the initial step for identification and classification of a large set of objects (RT.9). Signatures can comprise sounds, shapes, or other characterizing marks (such as color, material, and more) that can provide a sort of unique identifier for a product type. Once an instance is identified, some more information can be acquired by crawling the internet. The object is now recognized and more information about its behavior in the environment can be collected, organized, analyzed and reused. This can lead to a bottom up and incremental modeling of objects operating in a specific environment (ref. RT.4). Fig.5.2 shows the GPS architecture proposed in [64]. With this platform, data about vehicles passing by, meteo, pollution and other data were captured and organized in a way to describe the "environment" (a road), objects operating on it (their actions), and the con-

sequences of their behavior. Signatures were "calculated" by means of AI edge devices (Google's Coral boards) equipped with low cost environmental sensors and AI algorithms.

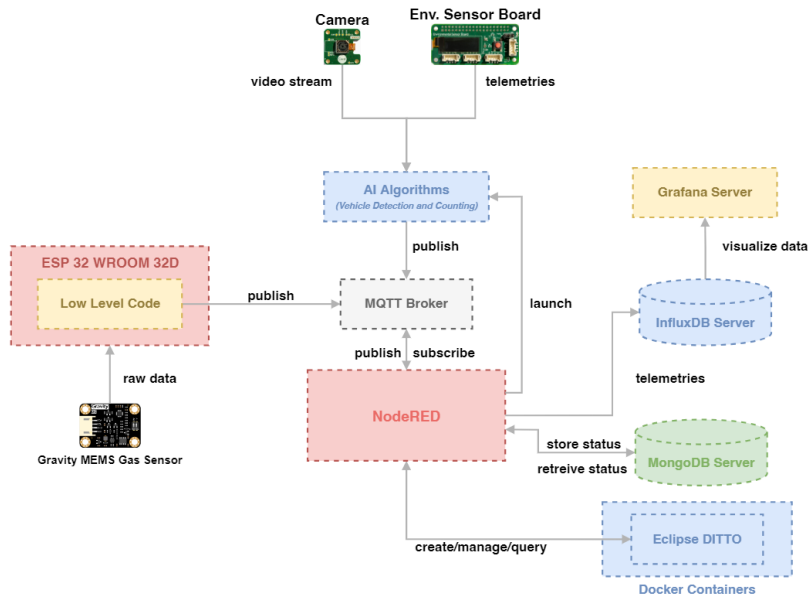


Figure 5.2: GPS System for Signatures and DTs Creation.

This Thesis [64] advocates the execution of identification and classification algorithms at the edge and provides experimental measurements related to edge vs. cloud processing of video frames. Edge processing saves bandwidth and has a good response time in terms of relevant data transfer to services and applications. The document also analyzes the capability of edge boards to execute AI algorithms for classification of vehicles and objects.

5.2.1 Opportunistic Digital Twin

These results were instrumental to introduce the concept of Opportunistic DT [32]. This is an "opportunistic interpretation of DT, which advances the traditional concept and provides a valid support for enabling next-generation solutions in dynamic, distributed and large scale scenarios as smart cities. Indeed, by collecting simple data from the environment and by opportunistically elaborating them through AI techniques directly at the network edge (referred as Edge Intelligence, RT.21), a digital version of a physical object can be built from the bottom up as well as dynamically manipulated and operated in a data-driven manner, thus enabling prompt responses to external stimuli and effective command actuation. Opportunism is the exploitation of any possibility (in terms of available data and algorithms for their treatment) that allows logical entities to progressively aims at their goals (e.g., reflection of the physical object) with more precision. This means to collect data in a wide manner (i.e., general purpose sensing) more than in a selective and specialized way, and exploit AI capabilities to transform the acquired data into information or understanding actions performed by physical objects. The ability to process data at the edge guarantees that data are locally processed and they are not transported or distributed in a larger environment.

The idea is to exploit signature characterization of objects and to derive additional information (descriptive but also prescriptive) from objects operating in the environment. In this way, the DT modeling is not only carried out in a top-down fashion, but is it left for further extension in a bottom-up way. The DT logic can "try" different data analysis or transformations to extract objects' information that is "evident" in the environment. New algorithms can be tried on these data, and, if they show utility, they can be consolidated and made part of the DT "model". Automata based analysis [90], rule and case based reasoning [91]

or other techniques can be enablers for this research (RT.3, RT.4). Some platform properties should support opportunistic processing: enough computing power at the edge; a vast general purpose sensing infrastructure to collect data; a set of well defined and tuned algorithms capable of transforming data into information; an open environment that supports the rapid introduction of new algorithms or methods or agents for transforming data. The difference with classic DTs is shown in Fig.5.3. Interplay between Top-Down and Bottom-

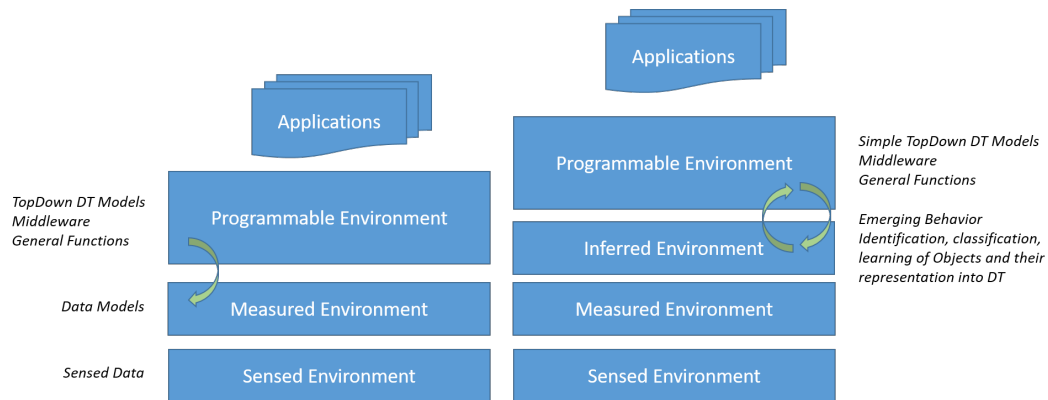


Figure 5.3: General Purpose and Synthetic Sensing as Enablers of Opportunistic DT

Up modeling is required to incrementally improve and tune-up the DT’s ability to catch the essential data that correspond to goals and aims of modeling. As a further research (RT.26), this approach could exploit the capability to train AI in the cloud and deploy it at the edge attempting to extract as much as possible information from the environment by means of synthetic sensing. The opportunistic DT plays a role to understand what are the limits of GPS (RT.6) in terms of capability to increase environment’s representation by exploiting environment measurements. The opportunistic architecture should offer flexibility and the ability to (download and) execute specific AI algorithms. The proposed architecture is a generalization of the one shown in Fig.5.2 with additional mechanisms to support the ability to launch and control new algorithms for data transformation. The choice and organization of new algorithms is retained in the cloud (and after some verification, they can be deployed at the Edge), because it has more computing resources for training and tune up. There is the possibility to observe different local environments and their commonalities or differences. Fig.5.2 depicts an initial architecture configuration. It uses some simple intelligence, represented by the logic of NodeRED, to activate new data transformation by means of new algorithms locally deployed in edge nodes. The same approach has been pursued in [65] with an application to a Traffic Monitoring System. This proved that the combination of GPS, Synthetic Sensing and AI is instrumental to the implementation of Opportunistic Digital Twins.

5.2.2 Looking for Bottom Up Models

Aiming at an approach consolidation and extension to video capture capabilities, another experiment has been conducted. Video was taken from the Shinjuku crossroad footage ¹, data extracted using YOLO detection capabilities, and traffic trajectory predictions determined by specific algorithms. The complete results of these works are documented in

¹available at <https://www.youtube.com/watch?v=6dp-bvQ7RWo>

Master Thesis: [92], and [93]. The original purpose was to track as many as possible objects in a complex environment like the Tokyo crossroad and to determine different flows of information. One first problem was to determine reference coordinates of the environment from the perspective of fixed camera. The solution was to refer to visual coordinates of video capture (expressed in pixels). This was an approximation, but was a simple method. Once a reference system was defined, objects could be identified and placed in the coordinate system. Then algorithms were trained (on capable servers) and trained YOLO models deployed in edge devices (TPU), in a way to exploit and assess a possible deployment at the edge for a smart city. Two issues were tackled during experiments: the difficulties of Multi-Object-Tracking in a complex environment, and the inference timing due to the limited (with respect to cloud and powerful GPU implementations) computing resources. After some "trail and error", it resulted that the identification and classification of vehicles was good, while tracking of pedestrians was quite complex and difficult to achieve due to two main factors: the reference coordinates for far apart points in the video, and the large amount of people crossing the roads all at once. It was decided to pursue the effort focusing only on optimization of vehicle tracking and to exploit these data for trajectory prediction. The method proposed in [94] was adopted. It is based on a definition of a grid (this is why a reference system was important) in which rows represent different lanes around the vehicle² and columns are cells of 4.6 meter each (representing an average vehicle size). This grid is used to identify and represent other vehicles close to the target one. Each vehicle is placed in the right position with respect to the target one and the grid represents a set of constraints. An LSTM was trained on the data collected from YOLO in the chosen Tokyo's crossroad. The position of vehicles referred to the visual coordinates of the video footage. Two different LSTM practices were used: an historical one (HIST) in which a vector represents the previous five positions of the vehicle plus the current location; and another (NBRS, i.e., neighbors) that considers the coordinates of vehicles classified as "close" to the target one in previous five time periods and in the current instant. The two models were implemented with PyTorch and Google Collab was used for processing. The results were encouraging, but still not precise. The HIST model, based on historical information, is capable of a good prediction of the target vehicle's coordinates while the trajectory prediction should be improved. The NBRS model has proved less effective and needs a rethinking to be useful. All the experiments' results are fully reported in [93]. As a forethought on trajectory prediction, the considered models were essentially developed and used on highways and fast roads. A urban environment is much more challenging for trajectories prediction. This should be taken into account in assessing the achieved results. As another possible research path, improved models and techniques for urban traffic prediction and trajectory should be considered and developed.

5.3 Reusing Data in a Smart City

This section reports attempts conducted in smart cities for evaluating GPS applicability.

5.3.1 Traffic Intensity and Pollution

Understanding and evaluating the relationship between traffic and pollutants, seen as General Purpose Sensing measurements, are the purposes of [95]. The Madrid data sets are

²The authors were using three lanes: left, center and right according to highways topology considered and represented in their data set.

considered and the approach of [29] is further consolidated and improved. The topical improvements are: a larger analysis of correlations between traffic intensity (Pearson and Spearman), pollutants, and meteorological conditions; a more precise analysis of traffic sensors (mobile or fixed) and their positions with respect to pollution stations; a larger number of ML/DL algorithms used, namely Long Short-Term Memory (LSTM), Gated recurrent units (GRUs), Bidirectional LSTM (BiLSTM), and a Convolutionary approach by introducing a Conv1D layer in the analysis of temporal data and the combination of LSTM and Conv1D. The data were organized in a curated dataset covering the entire year 2021 for selected measures (traffic intensity, relevant pollutants and meteorological conditions). Fig.5.4a represents the set of features selected for data analysis. Consistent tuning and

Features	Measure Units
CO	mg/m ³
NO	mg/m ³
NO ₂	mg/m ³
NO _x	mg/m ³
Traffic Flow	Vehicles/hour
Temperature	C°
Humidity	%
Wind Speed	Km/h
Wind Direction	Degrees
Pressure	mb

(a) Madrid's Features Selection

Model	MSE	MAE	RMSE
LSTM	0.025	0.126	0.159
GRU	0.020	0.111	0.142
BiLSTM	0.027	0.130	0.163
Con1D	0.021	0.115	0.146
LSTM_Conv1D	0.017	0.104	0.132

(b) Accuracy Achieved by Different Algorithms

Figure 5.4: Analysis of Madrid Data for correlating Traffic Intensity and Pollutants

training of algorithms have been carried out and the prepared dataset has been used for this purpose, and for validation and test. The different algorithms' results are reported in Fig.5.4b. With respect to Synthetic Sensing achievements, this research activity showed that different algorithms are important for determining accuracy, precision and quality of results. This means that, once identified the possibility of deriving the wanted information, the choice of best algorithm is a matter of optimization. From the standpoint of synthetic sensing, instead, it is important to figure out what data to collect, and what information can be derived from the general purpose sensed data. This is an example of the Information Density problem (RT.2). Determining how to curate and aggregate the data is a lesser problem than "what general data" to collect. From a bottom-up perspective, it is important to understand what data can provide the higher capability to infer new information. For the time being, there is not a theory or heuristic for deciding what to collect. And this is a relevant research topic further discussed in 6.4.

5.3.2 Traffic Intensity from Google Data

Another example of how data fusion can support general purpose and synthetic sensing is presented in [72]. Small-medium towns do not have the capability to deploy an extensive infrastructure for traffic monitoring. So they use different data sources such as Tom-Tom generated source-destination data and measures. In this experiment, data taken from Google's distance matrix were used³, but they were fused with other measurements. Two additional features were used: RATP's Bus real-time positioning dataset, and meteorological information. The novel idea was to add information about traffic as detected by specific bus rides in congested locations of the area under scrutiny. For completeness also meteo

³<https://developers.google.com/maps/documentation/distance-matrix/overview>

information were used to understand if they have impacts on the usage of vehicles. Several ML/DL algorithms have been tried (e.g., the Facebook Prophet) to finally converge towards the usage of LSTM. In this experiment, data are queried from different real-time sources and curated into a common FIWARE data model. They are used to train the chosen ML algorithms and for testing / validation of predictions. Several combinations of data were compared: only traffic data (i.e., Google ones); traffic and buses (Google and RATP ones); traffic and meteo; and traffic, buses and meteo. This approach explores when and how an increase in "feature" fusion influences the quality of predicted values for traffic. Table 5.1 shows how the different algorithms compare each other in term of Mean Average Error (MAE) in two periods (6 months and 1 month). This shows that, due to the nature of data

Table 5.1: Comparison of Performance Metric for Predicted Models

Error	MAE(FBProphet)	MAE(NeuralProphet)	LSTM
MAE (6months)	0.244	0.208	0.0502
MAE (one month)	0.19	0.16	0.12

(time series), LSTM was providing better results. Table 5.2 reports how the same algorithm (LSTM) performed with respect to the fusion of different data. The combination of all the

Table 5.2: LSTM Model Performances in terms of MAE, MSE and RMSE

Error Metric	MAE	MSE	RMSE
LSTM (Traffic)	0.12	0.028	0.17
LSTM (Traffic+Weather)	0.064	0.0086	0.093
LSTM (Traffic+Bus)	0.060	0.006	0.078
LSTM (Traffic+Bus+Weather)	0.056	0.006	0.070

considered features gives the best results, however traffic and bus delay aggregation gives a very accurate prediction that only slightly differs from the best one. Table 5.2 shows that "traffic + bus" contribute the most to prediction improvement with respect to the baseline usage of traffic. Also this result addresses the need to better understand how different features contribute the most to prediction accuracy. Some works already evaluate the single feature contribution to the final prediction's accuracy [96]. This is a starting point for determining how much increase in "information density" a feature can offer in a general purpose sensing approach (ref. section 6.4 and RT.2).

5.3.3 From Redundancy to Optimization

Another subject that emerges from Madrid data analysis is the large amount of sensors used for monitoring the environment. Are all these traffic intensity sensors needed? Can a reduction being such that the information content is the same? These are research issues discussed in [97]. Madrid sensors are deployed in all city's districts. To tackle some of the questions posed before, an analysis takes into consideration District 19 with the lowest number of sensors: 45. The goal is to study the correlation between devices and to use only one among correlated sensors as pivot for real-time measurement and to predict by means of LSTM the values of others (that can be considered redundant). Data from District's sensors for the month of October 2023 were collected, curated and normalized to create a well

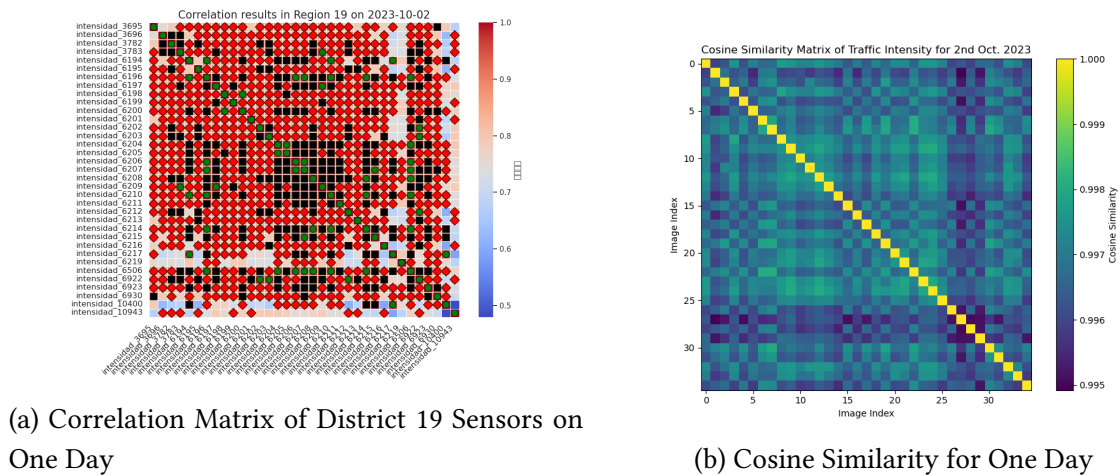


Figure 5.5: Finding the Correlation between District 19 Sensors.

formed dataset. The suitable sensors were 34 (others have missing or corrupted data). A normalization is done to focus on the trend of values and comparing the fluctuation between sensors' value independently from the max and min. At first, the Pearson correlation between pairs of sensors was computed. Fig.5.5a shows the correlation between all District19's sensors in a specific day. Red markers represent correlations ranging from 0.8 to 0.9, black markers indicate correlations between 0.9 and 0.95, and green markers highlight sensor pairs with Pearson correlation coefficients exceeding 0.95. This process was repeated for October 2023 considering day by day, weekday and month correlations. The second step is determining the similarity between sensors time series. Two distinct similarity measures are used to identify sensors with akin behavioral patterns: cosine similarity and deep learning-based similarity measure. Euclidean distance, perceptual hashing, and the Structural Similarity Index Measure (SSIM) were tried, but cosine similarity emerged as the most suitable. A calculation of cosine similarity between pairs of selected sensors over an entire days (with data collected every 15 min), week days (with values calculated over the hour); and the entire month (with values calculated as average per day) was performed. In this way it was possible to create a matrix figure representing the cosine similarity as showed in Fig.5.5b. Since the similarity score were very close (between 0.98 and 1), the next step was to create normalized histograms of values vector and to provide them as input to a Visual Geometry Group, VGG16, Neural Network[98]. The used VGG16 network comprises 13 convolutional layers and 5 Max Pooling layers, followed by convolutional layers to reduce the dimension and parameters' number of feature maps created by each convolution step. This produces a more viable range of similarity comprised between (0.60, 1.00). This process helps in identifying pivot sensors to use for determining values of correlated ones. These could be considers "idle / redundant" and used as backup in case of anomalies or other events. Fig.5.6a represents a network of correlated sensors. Once the pivots have been identified, they are used to predict, by means of an LSTM algorithm, values of other target sensors. Fig.5.6b shows the R^2 Score and MAE results obtained from sensor 6506 and its correlated sensors. Some considerations on this methodology:

- The method should be tested on a larger scale. Sensors in other Districts can be highly correlated, there is a need to keep track of this inter-District similarity.
- The correlation and similarity computation is time-consuming and for a large number

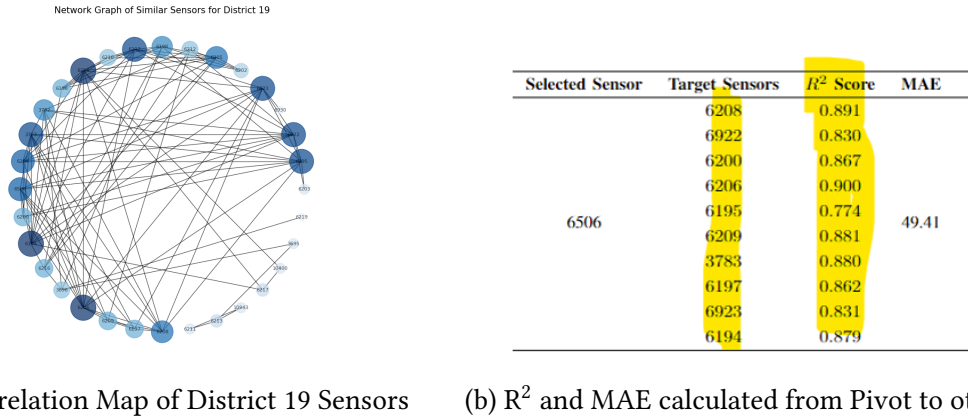


Figure 5.6: Correlation Map and LSTM Prediction Results for correlated Sensors

of sensors could be expensive.

- An algorithm for placing newer sensors in nearby Districts avoiding high level correlation should be devised (RT.27).
- Redundancy can be useful in case of anomalies or local events, but the costs and investments for a very dense sensor network may not be worth (RT.27).
- There is the need of a study about how local anomalies can reverberate on nearby sensors and how to calculate the traffic intensity without the deployment of sensors in the specific spot.

This study is another contribution to the topic of information density problem in an environment (RT.2). Its generalization could provide interesting hints and ideas for better understanding how to fully measure an environment with a reduced set of sensors.

5.4 An Architecture for General Purpose Sensing

Experiments have contributed to a continuous update of General Purpose and Synthetic Sensing architecture. The need for a coordination point for deployment and activation of new algorithms transforming general purpose data into information clearly emerged: this is why a centralized cloud based infrastructure capable of monitoring and policing the activities of edge nodes is introduced. It decides policies to download and activate them to receive the needed data. Cloud nodes can have the ability to more effectively train and tune-up algorithms and functions respect to edge nodes. When new functionalities are ready and useful, they could be "enforced" in the local edge nodes and results received by the cloud functions. A Policy Decision - Policy Enforcement - Policy Execution framework can be considered for implementing the GPS approach. A simple hierarchical architecture is shown in Fig.5.7. It depicts possible internal characterizations of cloud and edge nodes. With this organization, distinct nodes can apply different policies depending on local situation, location and scope of the GPS node. Intermediate edge nodes can execute functionalities or collecting data to be further processed at a regional or intermediate level. In addition, they can be used to manage and ensure the correct functioning of the deployed infrastructure at the edge. In certain cases (e.g., edge node malfunctioning, breaks in communications, and the like), they could also implement "virtual sensing" capabilities (i.e., generating data that reflect the typical data collected by the end node). Another possibility is the ability to create a real distributed infrastructure in which the FarEdge nodes are able

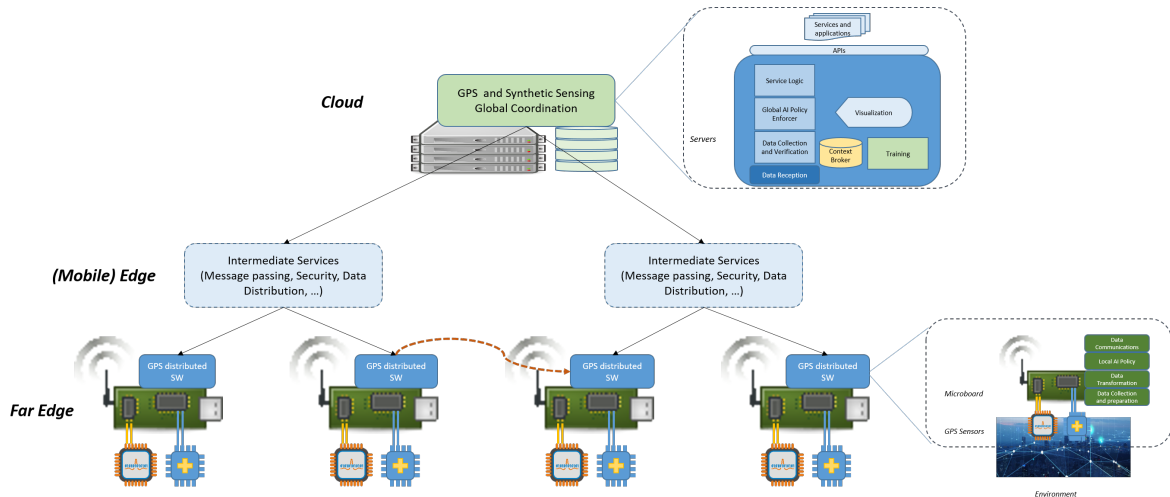


Figure 5.7: Distributed Architecture for General Purpose and Synthetic Sensing

to coordinate themselves (the red dotted line between central Far Edge Nodes in Fig.5.7), to determine, negotiate, and distribute the execution of policies in the entire system. This solution is interesting from the perspective of a fully decentralized GPS system. The Policy Enforcement infrastructure has been proposed and used in the Internet of Things domain [99], and, even more appealing and important, in the federated learning domain by DeepMind and Google [100]. GPS Federated Learning (RT.25) is an interesting research path that can lead to relevant contributions in several fields. Additional steps of consolidation are, indeed, needed to achieve a fully operating GPS architecture. The policy enforcement framework, currently, seems a good approach. It is simple, well known and effective. It should be tried and consolidated in the near future. Another essential aspect of a GPS architecture is the need for a large Repository of signatures (RT.9). A specific framework for building, hosting, compare signatures and provide information about the identified objects would be an added value that could promote a large use of signature based approach. This could be a sort of repository to incrementally build and consolidate with signatures captured with GPS and classified in several ways. As noted in this chapter, many convolutionary approaches are extensively used comparing "figure" representing metrics each other. This is another enabler for the creation of a signature based platform.

Part III

Something New

FUTURE RESEARCH PATHS

The future is no more uncertain than the present

– Walt Whitman

6.1	Bottom Up Digital Twins	58
6.2	Digital Twin and Explainable AI	58
6.3	Situation Awareness: Back to Reasoning	59
6.4	How Much Information from Sensing	60
6.5	A Possible Roadmap	61

This section presents descriptions of newer research challenges together with actual and future outcomes expected from starting or on-going activities.

6.1 Bottom Up Digital Twins

Building Digital Twins means incrementally model, design, and implement detailed aspects of a complex physical object. Specifications progressively lay out a prescriptive model and use descriptive models to align behavior with available data. Incremental detailing of prescriptive and descriptive models brings to a full specification that can be implemented. This is particularly valid for new systems or products. What happens if an existing complex object, a large one as a City or an organization, has to be modeled without a previous design or even with a shallow knowledge of its behavior?

Problem Statement: Deriving the Behavioral Model. Behavioral and prescriptive models are usually seen as synonymous. However, a prescriptive model is defined by PO designers, while a behavioral model emerges from actual actions and events executed by the PO. Designing a prescriptive model is a top-down activity, while identification of a behavioral model is bottom-up. It is based on event observation, and the inferring of rules and constraints that explain why and how these actions occur. The prescriptive modeling relates to programming and implementation from scratch of a system, while the behavioral modeling deals with the study of complex systems and their formalization by means of formal automatic models. DT research has partially incorporated results of bottom-up modeling through automata, agents and other formal modeling means [101], [102].

Proposal. Behavioral model of complex PO can be incrementally built and improved by means of observation (RT.4). A pragmatic objective is behavioral model construction by starting from well-structured templates (RT.16) and progressively adapting them to actual PO's behavior. For capturing rules and actions, automatic formal modeling or other mathematical means can be used. Availability of behavioral templates of system types enables specialization, improvement, and tune-up of model with actual data. Generative AI can create virtual critical situations that help to shape the wanted behavior of the DT. Some experiments, based on concepts introduced in [7] and [32], have been and are actually conducted to verify and support this approach. The research continues by defining templates and prove them with use cases.

Expected Results and Way Forwards. The outcome is a DT framework (RT.1) supporting incremental specification of a system's behavior by identifying environment features, actors acting upon it and actions. The method will determine formal representations of observed course of actions. The DT framework is expected to help in creating a granular behavioral model by recognizing additional actors operating in it (by signatures) or to crawl information about identified actors and extending the behavioral rules to encompass actions and events generated by these actors. This approximation can be extended as much as needed: it is a heuristic approach converging towards the actual behavior of the system. Generative AI mechanisms for studying critical situations will be implemented and tested.

6.2 Digital Twin and Explainable AI

The extensive usage of AI capabilities for predicting results of a DT will progressively approach better accuracy. For new situations, additional training and techniques will be needed. In this way, however, no explanation about reasons of actions will be provided. Initial attempts (RT.8) to use modeling capabilities of DT to explain the behavior of a PO

[103],[104] are emerging. A prescriptive model can programmatically describe causes and needs for certain actions. It is a means to provide explanations for DT behavior. So, *what is the actual problem?* Models supporting DTs of large or complex systems are approximations of reality and they leave margins to uncertainty. If there is not a well formulated and complete prescriptive model, then uncertainty grows and ML/DL comes to help for predicting the most likely results of the PO actions, but this introduces opacity.

From Predictive Modeling to Explanation Framework. The challenge is to prove that the need for explanation mechanisms is, in essence, "directly proportional" to the usage of predictive mechanisms and "inversely proportional" to the formal completeness of a prescriptive model. In other words, the more the prescriptive model is accurate, the more it is usable to explain DT's actions in the current situation.

Proposal. The envisaged research proposition is to develop prescriptive mechanisms that, even with errors due to uncertainty, can be applied to calculate expected results under initial conditions and explain why they occurred. These mechanisms help in tuning the prescriptive model by comparing the expected results with validated (predicted and actually occurring) ones over time for finding out actions, a behavioral scenario, that fit with actual data. An analysis of those scenarios can lead to the improvement of formal descriptions for calibrating the model's results with real ones. Behavioral scenarios can provide explanations for the achieved data.

Expected Results and Way Forwards. Explainability goals fits with the design and validation stage of modeling development. In that phase, it is up to humans to identify functions, input, general rules, constraints, and acceptable approximation of the formal model. In a second step (at the operation phase), an automatic tune-up of some parameters identified in the model can take place. This should result in a sort of incremental Case Based Models (or an Expert System made of rules and constraints) capable of relating events and actions to causes in a way intelligible to humans. The goal is to make the DT framework capable of choosing (or helping to do so) the right scenarios and to compare them with data (actual or predicted ones) by: demonstrating that the completeness of prescriptive models is a means to corroborate and explain results; automatically improving the quality of prescriptive model (in terms of formal functions or deterministic computational steps) to close the gap with actual behavior (and data) of the PO.

6.3 Situation Awareness: Back to Reasoning

Situation awareness is the ability to understand system status (= environment + actors behaving in it), and the actual course of actions; and the ability to act upon the system to keep it in an optimized condition, or, at least, in acceptable ones. Thresholds on important states help to determine critical situations. For a DT, situation awareness is detecting critical states and proactively governing system transitions towards acceptable states (RT.10). The goal is to support situation awareness and design it for DTs.

When There Are Not Datasets, When Situation Is New, Then Reason! All DT types should detect critical situations and warn about its critical level. Reactive and Autonomic DTs are concerned with intervention to mitigate the situation. Autonomic ones immediately put in place actions to mitigate the situation, while Reactive ones "suggest" possible actions. But how to achieve required level of situation awareness? The problem has two aspects: cause identification, how to determine important factors (in terms of states values) causing the critical situation; and cause mitigation, how to determine appropriate actions

that can solve or mitigate consequences of the situation. This pushes towards the application of reasoning mechanisms within the DT logic. Different strategies can be applied, but the focus of this research is on reasoning mechanisms to identify and cope with critical situations. Independently from the approach, there are some major problems to tackle: the knowledge representation of system and its "situation" [105]; application of the best methodology for understanding the causes; and identification of solutions.

Proposal. A DT framework must offer as built-in functions situation identification, classification, warning and reaction (for Reactive and Autonomics DTs). The research activity focuses on how to provide these functions. The prerequisite is to collect and represent an actionable knowledge about the situation. Based on that, different techniques can be applied to find causes. The solution phase (i.e., the selection of actions that can lead back the system in a non-critical state) needs to evaluate different courses of actions and select the one that ensure the most effective results. This can be done in two phases: mitigation actions and recovery plans. This activity tries to exploit prescriptive models and generative AI. These features can be used to create extreme situations that put the DT in critical states: different logics can be determined and tried to sort out the simulated issues. This allows to create a number of realistic use cases that can be re-used at run-time and can help in generating a better representation and reaction of the situation.

Expected Results and Way Forwards. The aim is to understand how to detect real issues and develop mechanisms for implementation of Reactive or Autonomic DTs capable of reasoning on critical situations (situation aware). It is expected to gain experience and insights on different problem domains (e.g., industrial critical systems like nuclear power stations can exhibit different needs respect to a Smart city). The major outcome is to identify and use the most appropriate technologies for describing critical situations and reasoning about them considering and coping with acceptable risk levels for different problem domains.

6.4 How Much Information from Sensing

General purpose and synthetic sensing pose research challenges (RT.6, RT.7): is it possible to "measure" the capability of information extraction of a general purpose sensing infrastructure? Is it possible to define a theory/methodology to evaluate what is the smallest possible set of sensing capabilities needed to fully represent a system with respect to an applications? Is it possible to measure how much information can be derived/inferred by the actual/current system information? These questions represent investigations related to the **density of information** problem (RT.2).

Problem Statement: When Information Is Just Enough. In paper [97], the problem of sensors redundancy in Madrid was considered. A correlated problem is to define the relation between granularity and accuracy of monitoring a certain area (RT.27). A small number of sensors plus mathematical/physical models can provide the needed information to the large majority of applications. The information is then determined by applying models, algorithms to data collected by sensors and calculated in a specific spot of the environment. Other factors are: data fusion capability of increasing the usefulness of existing data in creating new information; and assessment of how general-purpose and synthetic sensing can offer the highest level of "information density" possible, and how to reach it (e.g., exploiting the expressiveness of sensing means: video vs. noise sensing).

Proposal. The objective is to define a methodology for determine and calculate "information density" in a system. Some issues are: how to "measure" capabilities and limits of the

sensing infrastructure to capture useful measures; how to evaluate the possibility to use these measures to infer new information through data fusion, transformation models, and inferring methods; how to calculate redundancy (or lack) of sensing in the environment. These issues dependent on the system taken into consideration, but generalizations can be pursued to define a methodology applicable to different problem domains. The definition of an "information density" theory is difficult, but, in case of results, this goal will be pursued. **Expected Results and Way Forwards.** Expected outcomes are several, they range from a simple methodology for helping practitioners in effectively deploy sensor infrastructures avoiding redundancy, up to a more accurate understanding of general-purpose and synthetic sensing limitations in specific problem domains. On a theoretical side, a better formalization of "information density" for a system is a needed step towards a more systematic methodology. The consequent definition of synthetic sensing expressive power in relation to actual sensed information is a valuable result that can help in progressing with developments of this approach.

6.5 A Possible Roadmap

Research activities are framed in a plan with academic goals and industrial follow-ups:

1. The incremental prototyping of a General Purpose and Synthetic Sensing architecture (see section 5.4, RT.6).
2. The construction of a Signature Architecture comprising a Repository for large collections of signatures with identification and classification of objects (RT.5, RT.9).
3. The prototyping of functions for a Integrated Development Framework with customizable behavioral templates (RT.23).
4. An extension of Network Digital Twin implementation improving the representation of Edge Cloud Continuum, the integration of network aspects with distributed computing, and a move towards in-network computing (RT.20).
5. The development of DT representations for Smart Cities (RT.17) aiming at: a) monitoring and optimization of resources; and b) planning of processes within the urban environment.
6. The definition of a sound and measurable problem statement related to the Information Density issue and its mathematical/heuristic formalization (RT.2).

These activities are characterized by implementations. The developments remain at the prototypes level, but, if industrial partnering is possible, implementations will be consolidated. They can contribute to National or European Projects. Fig.6.1 depicts a roadmap of research activities. Some of them have already started and Ph.D. students allocated and working on subject matters. Others are at their beginning and may not even have been fully planned or funded. The most consolidated ones are Smart City Digital Twin and Network DT. General Purpose Sensing Architecture requires more prototyping, identification of additional use cases, and a better definition of the bottom-up modeling approach. The Integrated Development Framework for DT is implementation oriented and has two facets: the construction of DT functions and components to be integrated into an existing Integrated Development Environment; the identification, implementation and assessment of prescriptive (top-down) and behavioral (bottom-up) models for DT representations.

Bibliography

The Signature Management Architecture can be considered as a software development for a research topic, but it can also be seen as a global *Shazam* for different types of signatures. Information Density Problem is a newer research activity that needs to be better formalized. A large State-of-the-Art, encompassing different problem domains, is the basis for further developments and decisions. The industrial exploitation (represented in Fig.6.2) comprises joint projects, specific Ph.D. funding or larger partnership if there is a business proposition.

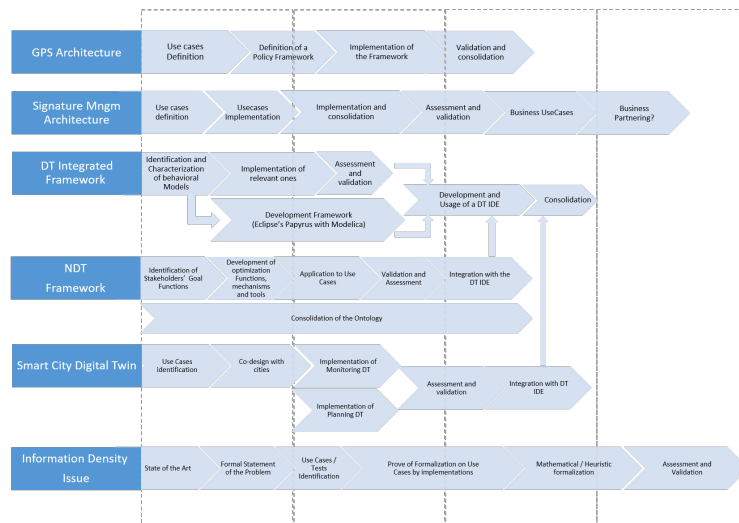


Figure 6.1: Research Roadmap for the identified Activities.

The activities for the creation of Smart City DTs have a dual exploitation: to focus on specific problem domains and create a DT "monitoring and optimization" model; to develop processes and planning capabilities. This helps City Planners in formalizing their needs and possibilities. The NDT potential partners are in the domain of

ICT enterprises, e.g., Communications Service Providers interested in a DT representation of Networks, or Cloud / Computing Infrastructure Providers interested to help customers to optimize consumption and usage of resources within an Edge-Cloud Continuum. The GPS Architecture can be used to optimize the deployment of costly infrastructure by preserving the ability to gather future needed data. It can be complemented by an effective analysis supported by the results of Information Density activities. This latter activity is the one with longest path to exploitation. Its potential value is in a meaningful contribution for understanding how to extract the most possible information exploiting available data. Pursuing these activities is challenging and requires many resources. Some issues may be harder than expected and take more time to be solved. This planning, if supported by capable skilled people, can be achieved and satisfy expected requirements for obtaining valuable Ph.D. results from Doctorates.

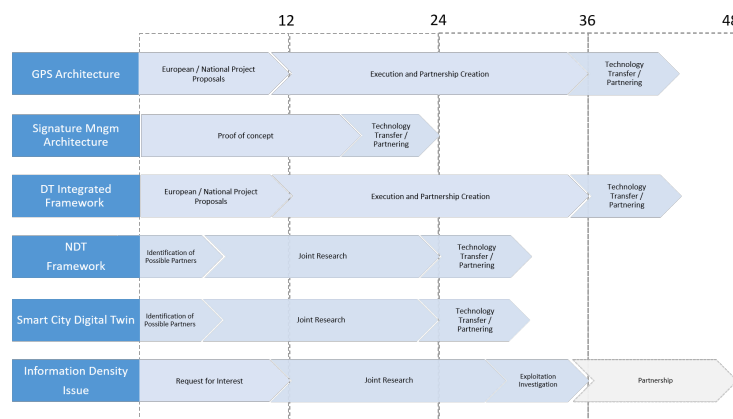


Figure 6.2: Possible Exploitation Paths.

BIBLIOGRAPHY

- [1] Kevin Parker. “Software still eating world.” In: *Plant Engineering* 74.7 (2020), pp. 5–6.
- [2] Grzegorz Blinowski, Anna Ojdowska, and Adam Przybyłek. “Monolithic vs. Microservice Architecture: A Performance and Scalability Evaluation”. In: *IEEE Access* 10 (2022), pp. 20357–20374. DOI: [10.1109/ACCESS.2022.3152803](https://doi.org/10.1109/ACCESS.2022.3152803).
- [3] Daniel Balouek-Thomert et al. “Towards a computing continuum: Enabling edge-to-cloud integration for data-driven workflows”. In: *The International Journal of High Performance Computing Applications* 33.6 (2019), pp. 1159–1174.
- [4] Shadi Al-Sarawi et al. “Internet of Things Market Analysis Forecasts, 2020–2030”. In: *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4)*. 2020, pp. 449–453. DOI: [10.1109/WorldS450073.2020.9210375](https://doi.org/10.1109/WorldS450073.2020.9210375).
- [5] Amir Reza Jafari et al. “Data enrichment toolchain: A use-case for correlation analysis of air quality, traffic, and meteorological metrics in Madrid’s smart city”. In: *Internet of Things* 26 (2024), p. 101232. ISSN: 2542-6605. DOI: <https://doi.org/10.1016/j.iot.2024.101232>. URL: <https://www.sciencedirect.com/science/article/pii/S2542660524001732>.
- [6] Gierad Laput, Yang Zhang, and Chris Harrison. “Synthetic sensors: Towards general-purpose sensing”. In: *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*. 2017, pp. 3986–3999.
- [7] Roberto Minerva, Faraz Malik Awan, and Noel Crespi. “Exploiting digital twins as enablers for synthetic sensing”. In: *IEEE Internet Computing* 26.5 (2021), pp. 61–67.
- [8] Michael W. Grieves. “Digital Twins: Past, Present, and Future”. In: *The Digital Twin*. Ed. by Noel Crespi, Adam T. Drobot, and Roberto Minerva. Cham: Springer International Publishing, 2023, pp. 97–121. ISBN: 978-3-031-21343-4. DOI: [10.1007/978-3-031-21343-4_4](https://doi.org/10.1007/978-3-031-21343-4_4). URL: https://doi.org/10.1007/978-3-031-21343-4_4.
- [9] Shaun West et al. “Value propositions enabled by digital twins in the context of servitization”. In: *Advanced Services for Sustainability and Growth* (2020), p. 152.
- [10] Roberto Minerva, Charalampos Z. Patrikakis, and Tiziana Margaria. “Digitally Cloning the Physical”. In: *IT Professional* 24.6 (2022), pp. 18–19. DOI: [10.1109/MITP.2022.3224263](https://doi.org/10.1109/MITP.2022.3224263).
- [11] Noel Crespi, Adam T. Drobot, and Roberto Minerva. *The Digital Twin*. Springer International Publishing, 2024. ISBN: 9783031213458.

Bibliography

- [12] Noel Crespi, Adam T Drobot, and Roberto Minerva. “The Digital Twin: What and Why?” In: *The Digital Twin*. Springer, 2023, pp. 3–20.
- [13] Rogardt Heldal et al. “Descriptive vs prescriptive models in industry”. In: *Proceedings of the acm/ieee 19th international conference on model driven engineering languages and systems*. 2016, pp. 216–226.
- [14] Adam Thelen et al. “A comprehensive review of digital twin—part 1: modeling and twinning enabling technologies”. In: *Structural and Multidisciplinary Optimization* 65.12 (2022), p. 354.
- [15] Luis Sánchez et al. “Data Enrichment Toolchain: A Data Linking and Enrichment Platform for Heterogeneous Data”. In: *IEEE Access* 11 (2023), pp. 103079–103091. DOI: [10.1109/ACCESS.2023.3317705](https://doi.org/10.1109/ACCESS.2023.3317705).
- [16] Azad M Madni and Michael Sievers. “Model-based systems engineering: Motivation, current status, and research opportunities”. In: *Systems Engineering* 21.3 (2018), pp. 172–190.
- [17] Ryan B Walton, Frank W Ciarallo, and Lance E Champagne. “A unified digital twin approach incorporating virtual, physical, and prescriptive analytical components to support adaptive real-time decision-making”. In: *Computers & Industrial Engineering* 193 (2024), p. 110241.
- [18] Lu Wang et al. “Explainable modeling in digital twin”. In: *2021 Winter Simulation Conference (WSC)*. IEEE, 2021, pp. 1–12.
- [19] Kazuma Kobayashi and Syed Bahauddin Alam. “Explainable, interpretable, and trustworthy AI for an intelligent digital twin: A case study on remaining useful life”. In: *Engineering Applications of Artificial Intelligence* 129 (2024), p. 107620.
- [20] Noel Crespi, Adam T Drobot, and Roberto Minerva. “The Digital Twin in action and directions for the future”. In: *The Digital Twin*. Springer, 2023, pp. 1201–1217.
- [21] Fei Tao et al. “Digital twin modeling”. In: *Journal of Manufacturing Systems* 64 (2022), pp. 372–389.
- [22] Javier Conde et al. “Modeling digital twin data and architecture: A building guide with fiware as enabling technology”. In: *IEEE Internet Computing* 26.3 (2021), pp. 7–14.
- [23] Shyam Varan Nath, Pieter Van Schalkwyk, and Dan Isaacs. *Building industrial digital twins: Design, develop, and deploy digital twin solutions for real-world industries using Azure digital twins*. Packt Publishing Ltd, 2021.
- [24] Salvatore Cavalieri and Salvatore Gambadoro. “Proposal of mapping digital twins definition language to open platform communications unified architecture”. In: *Sensors* 23.4 (2023), p. 2349.

- [25] Marcus Vinicius Pereira Pessoa, Ana Paula Lopes Schuch, and Juliana Melo Bezerra. “Using model-based systems engineering to design system-based digital twins”. In: *INCOSE International Symposium*. Vol. 33. 1. Wiley Online Library. 2023, pp. 1435–1449.
- [26] Stefan Vöth and Maria Vasilyeva. “Potential of Modelica for the creation of digital twins”. In: *Advances in raw material industries for sustainable development goals*. CRC Press, 2020, pp. 386–389.
- [27] Concetta Semeraro et al. “Data-driven invariant modelling patterns for digital twin design”. In: *Journal of Industrial Information Integration* 31 (2023), p. 100424.
- [28] Faraz Malik Awan et al. “A comparative analysis of machine/deep learning models for parking space availability prediction”. In: *Sensors* 20.1 (2020), p. 322.
- [29] Faraz Malik Awan, Roberto Minerva, and Noel Crespi. “Improving Road Traffic Forecasting Using Air Pollution and Atmospheric Data: Experiments Based on LSTM Recurrent Neural Networks”. In: *Sensors* 20.13 (2020). ISSN: 1424-8220. DOI: [10 . 3390/s20133749](https://doi.org/10.3390/s20133749). URL: <https://www.mdpi.com/1424-8220/20/13/3749>.
- [30] Roberto Minerva, Gyu Myoung Lee, and Noel Crespi. “Digital twin in the IoT context: A survey on technical features, scenarios, and architectural models”. In: *Proceedings of the IEEE* 108.10 (2020), pp. 1785–1824.
- [31] Roberto Minerva and Noel Crespi. “Digital Twins: Properties, Software Frameworks, and Application Scenarios”. In: *IT Professional* 23.1 (2021), pp. 51–55. DOI: [10 . 1109/MITP.2020.2982896](https://doi.org/10.1109/MITP.2020.2982896).
- [32] Claudio Savaglio et al. “Opportunistic Digital Twin: an Edge Intelligence enabler for Smart City”. In: *ACM Trans. Sen. Netw.* (Aug. 2023). Just Accepted. ISSN: 1550-4859. DOI: [10 . 1145/3616014](https://doi.org/10.1145/3616014). URL: <https://doi.org/10.1145/3616014>.
- [33] Mehdi Kherbache, Moufida Maimour, and Eric Rondeau. “Digital twin network for the IIoT using eclipse ditto and hono”. In: *IFAC-PapersOnLine* 55.8 (2022), pp. 37–42.
- [34] Bernd Schmitt. “From Atoms to Bits and Back: A Research Curation on Digital Technology and Agenda for Future Research”. In: *Journal of Consumer Research* 46.4 (Aug. 2019), pp. 825–832. ISSN: 0093-5301. DOI: [10 . 1093/jcr/ucz038](https://doi.org/10.1093/jcr/ucz038). URL: <https://doi.org/10.1093/jcr/ucz038>.
- [35] Stefano Mariani, Marco Picone, and Alessandro Ricci. “About digital twins, agents, and multiagent systems: a cross-fertilisation journey”. In: *International Conference on Autonomous Agents and Multiagent Systems*. Springer. 2022, pp. 114–129.
- [36] Abdulmotaleb El Saddik. “Digital twins: The convergence of multimedia technologies”. In: *IEEE multimedia* 25.2 (2018), pp. 87–92.

Bibliography

- [37] Roberto Minerva. “From Internet of Things to the Virtual Continuum: An architectural view”. In: *2014 Euro Med Telco Conference (EMTC)*. 2014, pp. 1–6. DOI: [10 . 1109/EMTC.2014.6996633](https://doi.org/10.1109/EMTC.2014.6996633).
- [38] David Jones et al. “Characterising the Digital Twin: A systematic literature review”. In: *CIRP journal of manufacturing science and technology* 29 (2020), pp. 36–52.
- [39] Roberto Minerva et al. “Artificial intelligence and the digital twin: An essential combination”. In: *The digital twin*. Springer, 2023, pp. 299–336.
- [40] Mica R Endsley and Daniel J Garland. *Situation awareness analysis and measurement*. CRC press, 2000.
- [41] Yevhen Burov. “Knowledge Based Situation Awareness Process Based on Ontologies.” In: *COLINS*. 2021, pp. 413–423.
- [42] Larry Schmitt and David Copps. “The Business of Digital Twins”. In: *The Digital Twin*. Springer, 2023, pp. 21–63.
- [43] Corrado Moiso and Roberto Minerva. “Towards a user-centric personal data ecosystem The role of the bank of individuals’ data”. In: *2012 16th International Conference on Intelligence in Next Generation Networks*. 2012, pp. 202–209. DOI: [10 . 1109 / ICIN.2012.6376027](https://doi.org/10.1109/ICIN.2012.6376027).
- [44] Roberto Minerva and Noël Crespi. “Unleashing the disruptive potential of user-controlled identity management”. In: *2011 Technical Symposium at ITU Telecom World (ITU WT)*. 2011, pp. 1–6.
- [45] Eric Brewer. “CAP twelve years later: How the " rules" have changed”. In: *Computer* 45.2 (2012), pp. 23–29.
- [46] Ralph Barthel et al. “An internet of old things as an augmented memory system”. In: *Personal and ubiquitous computing* 17 (2013), pp. 321–333.
- [47] Shaun West et al. “Value propositions enabled by digital twins in the context of servitization”. In: *Advanced Services for Sustainability and Growth* (2020), p. 152.
- [48] Junaid Ahmed. “Digital Twin, Servitization, Circular Economy, and Lean Manufacturing: A Step toward Sustainability”. In: *Human Factors in Engineering*. CRC Press, 2023, pp. 135–154.
- [49] Manoj Herath et al. “Advancements in Digital Twin: A Comprehensive Review of Challenges and Solutions in Modeling and Software Architectures”. In: *Software and Systems Modeling* (2024), pp. 1–35. Submitted.
- [50] Yuvraj Agarwal et al. *Virtual sensor system*. US Patent 10,436,615. Oct. 2019.
- [51] IoTNow. *IoT moves to large deployments*. Accessed on July 7th, 2024. March 1, 2024. URL: <https://www.iot-now.com/2024/03/01/143049-iot-moves-to-large-deployments/>.

- [52] Glenn A Wellbrock et al. “Explore benefits of distributed fiber optic sensing for optical network service providers”. In: *Journal of Lightwave Technology* 41.12 (2023), pp. 3758–3766.
- [53] Tong Meng et al. “A survey on machine learning for data fusion”. In: *Information Fusion* 57 (2020), pp. 115–129. ISSN: 1566-2535. DOI: <https://doi.org/10.1016/j.inffus.2019.12.001>. URL: <https://www.sciencedirect.com/science/article/pii/S1566253519303902>.
- [54] Jiaxin Li et al. “Deep learning in multimodal remote sensing data fusion: A comprehensive review”. In: *International Journal of Applied Earth Observation and Geoinformation* 112 (2022), p. 102926. ISSN: 1569-8432. DOI: <https://doi.org/10.1016/j.jag.2022.102926>. URL: <https://www.sciencedirect.com/science/article/pii/S1569843222001248>.
- [55] Faraz Malik Awan, Roberto Minerva, and Noel Crespi. “Using Noise Pollution Data for Traffic Prediction in Smart Cities: Experiments Based on LSTM Recurrent Neural Networks”. In: *IEEE Sensors Journal* 21.18 (2021), pp. 20722–20729. DOI: [10.1109/JSEN.2021.3100324](https://doi.org/10.1109/JSEN.2021.3100324).
- [56] Leif-Thore Reiche et al. “The digital twin of a system: A structure for networks of digital twins”. In: *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*. IEEE, 2021, pp. 1–8.
- [57] David McKee. *Platform Stack Architectural Framework: An Introductory Guide*. Tech. rep. Digital Twin Consortium, July 2023.
- [58] Armir Bujari et al. “A Digital Twin Decision Support System for the Urban Facility Management Process”. In: *Sensors* 21.24 (2021). ISSN: 1424-8220. DOI: [10.3390/s21248460](https://doi.org/10.3390/s21248460). URL: <https://www.mdpi.com/1424-8220/21/24/8460>.
- [59] Luis F Rivera et al. “On the engineering of IoT-intensive digital twin software systems”. In: *Proceedings of the IEEE/ACM 42nd International Conference on Software Engineering Workshops*. 2020, pp. 631–638.
- [60] TAO Fei et al. “makeTwin: A reference architecture for digital twin software platform”. In: *Chinese Journal of Aeronautics* 37.1 (2024), pp. 1–18.
- [61] Zoé Chevallier, Béatrice Finance, and Benjamin Cohen Boulakia. “A Reference Architecture for Smart Building Digital Twin.” In: *SeDiT@ ESWC 2020.5* (2020).
- [62] Alexandros Valantasis, Nikos Makris, and Thanasis Korakis. “Orchestration software for resource constrained datacenters: an experimental evaluation”. In: *2022 IEEE 8th International Conference on Network Softwarization (NetSoft)*. IEEE, 2022, pp. 121–126.

Bibliography

- [63] Syed Mohsan Raza et al. “Definition Of Digital Twin Network Data Model in The Context of Edge-Cloud Continuum”. In: *2023 IEEE 9th International Conference on Network Softwarization (NetSoft)*. 2023, pp. 402–407. DOI: [10 . 1109 / NetSoft57336 . 2023 . 10175444](https://doi.org/10.1109/NetSoft57336.2023.10175444).
- [64] Vincenzo Barbuto. *Creating Digital Twins with General Purpose Sensing: Architecture and Initial Experiments*. Master’s thesis. Universita degli Studi della Calabria – Rende (CS) Italy, June 2022. URL: [https : / / labs . dimes . unical . it / speme / wp - content / uploads / sites / 8 / Tesi _ Digital _ Twin _ Barbuto _ final - compresso . pdf](https://labs.dimes.unical.it/speme/wp-content/uploads/sites/8/Tesi_Digital_Twin_Barbuto_final-compresso.pdf).
- [65] Vincenzo Barbuto et al. “Towards an Edge Intelligence-Based Traffic Monitoring System”. In: *2023 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. 2023, pp. 3434–3439. DOI: [10 . 1109 / SMC53992 . 2023 . 10393907](https://doi.org/10.1109/SMC53992.2023.10393907).
- [66] Aung Kaung Myat. *Traffic Intensity Detection Using General-Purpose Sensors*. Master’s thesis. Évry-Courcouronnes, Essonne, France, 2023.
- [67] Aung Kaung Myat et al. “Traffic Intensity Detection Using General-Purpose Sensing”. In: *IEEE Sensors Letters* 7.11 (2023), pp. 1–4. DOI: [10 . 1109 / LSENS . 2023 . 3315251](https://doi.org/10.1109/LSENS.2023.3315251).
- [68] Roberto Minerva et al. *D1.1: Report on semantic features extraction for heterogeneous data sources contextualization*. Tech. rep. EU SALTED Project, 2022. URL: [https : / / salted - project . eu / wp - content / uploads / 2023 / 02 / D1 - Report - on - semantic - features - extraction - for - heterogeneous - data - sources - contextualization . pdf](https://salted-project.eu/wp-content/uploads/2023/02/D1-Report-on-semantic-features-extraction-for-heterogeneous-data-sources-contextualization.pdf).
- [69] Gilles Privat and Alexey Medvedev. *Guidelines for Modelling with NGSI-LD*. White Paper. ETSI, 2021.
- [70] Maira Alvi et al. “Traffic Flow Prediction in Sensor-Limited Areas through Synthetic Sensing and Data Fusion”. In: *IEEE Sensors Letters* (2024).
- [71] Víctor González et al. *D2.2: Report on data modelling and linking*. Tech. rep. EU SALTED Project, 2023.
- [72] Maira Alvi et al. “Traffic Flow Prediction in Sensor-Limited Areas through Synthetic Sensing and Data Fusion”. In: *IEEE Sensors Letters* (2024).
- [73] Paul Almasan et al. “Network digital twin: Context, enabling technologies, and opportunities”. In: *IEEE Communications Magazine* 60.11 (2022), pp. 22–27.
- [74] Yiwen Wu, Ke Zhang, and Yan Zhang. “Digital twin networks: A survey”. In: *IEEE Internet of Things Journal* 8.18 (2021), pp. 13789–13804.
- [75] Oscar Corcho et al. “A high-level ontology network for ICT infrastructures”. In: *The Semantic Web–ISWC 2021: 20th International Semantic Web Conference, ISWC 2021, Virtual Event, October 24–28, 2021, Proceedings 20*. Springer. 2021, pp. 446–462.

- [76] Syed Mohsan Raza et al. “Empowering Microservices: A Deep Dive into Intelligent Application Component Placement for Optimal Response Time”. In: *Journal of Network and Systems Management* 32.4 (2024), p. 84.
- [77] Manoj Herath, Roberto Minerva, and Noel Crespi. “Towards the definition of SysML Behavioral Templates for Digital Twin Modeling”. In: *2024 IEEE 10th World Forum on Internet of Things (WF-IoT)*. IEEE. 2024, pp. 1–6. rejected.
- [78] Maira Alvi et al. *D2.2 Technical Report on Platform Development and Deployment*. Tech. rep. EU Climaborough Project, 2024.
- [79] Peter Fritzson. “The openmodelica environment for building digital twins of sustainable cyber-physical systems”. In: *2021 Winter Simulation Conference (WSC)*. IEEE. 2021, pp. 1–12.
- [80] Roberto Minerva, Faraz Malik Awan, and Noel Crespi. “Exploiting Digital Twins as Enablers for Synthetic Sensing”. In: *IEEE Internet Computing* 26.5 (2022), pp. 61–67. DOI: [10.1109/MIC.2021.3051674](https://doi.org/10.1109/MIC.2021.3051674).
- [81] Madhusri Maity, Sriparna Banerjee, and Sheli Sinha Chaudhuri. “Faster r-cnn and yolo based vehicle detection: A survey”. In: *2021 5th international conference on computing methodologies and communication (ICCMC)*. IEEE. 2021, pp. 1442–1447.
- [82] Kan Wu et al. “Tinyvit: Fast pretraining distillation for small vision transformers”. In: *European conference on computer vision*. Springer. 2022, pp. 68–85.
- [83] Junting Pan et al. “Edgevits: Competing light-weight cnns on mobile devices with vision transformers”. In: *European Conference on Computer Vision*. Springer. 2022, pp. 294–311.
- [84] Sachin Mehta and Mohammad Rastegari. “Mobilevit: light-weight, general-purpose, and mobile-friendly vision transformer”. In: *arXiv preprint arXiv:2110.02178* (2021).
- [85] Andrew Howard et al. “Searching for mobilenetv3”. In: *Proceedings of the IEEE/CVF international conference on computer vision*. 2019, pp. 1314–1324.
- [86] Kaiming He et al. “Deep residual learning for image recognition”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.
- [87] Kai Han et al. “A survey on vision transformer”. In: *IEEE transactions on pattern analysis and machine intelligence* 45.1 (2022), pp. 87–110.
- [88] Takumi Kato and Ryuji Yokote. “Effect of driving sound of electric vehicle on product attractiveness”. In: *Human-Centric Intelligent Systems* 3.3 (2023), pp. 416–424.
- [89] Yang Zhao, Hua Dong, and Long Liu. “Sound Design for Electric Vehicles: Lessons and Requirements Learned from a Chinese Automobile Manufacturer”. In: *Advances in Industrial Design: Proceedings of the AHFE 2020 Virtual Conferences on Design for Inclusion, Affective and Pleasurable Design, Interdisciplinary Practice in Industrial Design, Kansei Engineering, and Human Factors for Apparel and Textile Engineering, July 16–20, 2020, USA*. Springer. 2020, pp. 537–543.

Bibliography

- [90] Mark Burgin. “Triadic automata and machines as information transformers”. In: *Information* 11.2 (2020), p. 102.
- [91] J Partanen. “Guiding urban self-organization: Combining rule-based and case-based planning”. In: *Environment and Planning B: Urban Analytics and City Science* 47.2 (2020), pp. 304–320.
- [92] Stefano Perna. *Riconoscimento e tracciamento di oggetti nelle smart city tramite general-purpose sensing ed intelligenza artificiale*. Master’s thesis. Universita degli Studi della Calabria – Rende (CS) Italy, 2023. URL: https://labs.dimes.unical.it/speme/wp-content/uploads/sites/8/Tesi___Stefano_Perna-1-1.pdf.
- [93] Fabrizio Mangione. *Un approccio basato su LSTM per la previsione delle traiettorie dei veicoli in ambito Smart City*. Master’s thesis. Universita degli Studi della Calabria – Rende (CS) Italy, 2023. URL: https://labs.dimes.unical.it/speme/wp-content/uploads/sites/8/Master_thesis_Fabrizio_Mangione_signed_pdf_a-1.pdf.
- [94] Nachiket Deo and Mohan M Trivedi. “Convolutional social pooling for vehicle trajectory prediction”. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*. 2018, pp. 1468–1476.
- [95] Stefano Morrone. *Previsione del traffico stradale tramite general-purpose sensing e machine learning*. Master’s thesis. Universita degli Studi della Calabria – Rende (CS) Italy, 2023. URL: https://labs.dimes.unical.it/speme/wp-content/uploads/sites/8/tesi_morrone_stefano-1.pdf.
- [96] Asad Ali and Carmine Gravino. “Evaluating the impact of feature selection consistency in software prediction”. In: *Science of Computer Programming* 213 (2022), p. 102715. ISSN: 0167-6423. DOI: <https://doi.org/10.1016/j.scico.2021.102715>. URL: <https://www.sciencedirect.com/science/article/pii/S0167642321001088>.
- [97] Leyuan Ding et al. “Deep Learning for Reducing Redundancy in Madrid’s Traffic Sensor Network”. In: *2024 IEEE 49th Conference on Local Computer Networks (LCN)*. IEEE. 2024, pp. 1–6.
- [98] Karen Simonyan and Andrew Zisserman. “Very deep convolutional networks for large-scale image recognition”. In: *arXiv preprint arXiv:1409.1556* (2014).
- [99] Sabrina Sicari et al. “Dynamic policies in internet of things: enforcement and synchronization”. In: *IEEE Internet of Things Journal* 4.6 (2017), pp. 2228–2238.
- [100] Matthew W Hoffman et al. “Acme: A research framework for distributed reinforcement learning”. In: *arXiv preprint arXiv:2006.00979* (2020).
- [101] Christopher W Weimer, John O Miller, and Raymond R Hill. “Agent-based modeling: An introduction and primer”. In: *2016 Winter Simulation Conference (WSC)*. IEEE. 2016, pp. 65–79.

- [102] Chukwudi Nwogu et al. “Towards a requirement-driven digital twin architecture”. In: *Procedia CIRP* 107 (2022), pp. 758–763.
- [103] Lu Wang et al. “Explainable Modeling in Digital Twin”. In: *2021 Winter Simulation Conference (WSC)*. 2021, pp. 1–12. DOI: [10 . 1109 / WSC52266 . 2021 . 9715321](https://doi.org/10.1109/WSC52266.2021.9715321).
- [104] Kazuma Kobayashi and Syed Bahauddin Alam. “Explainable, interpretable, and trustworthy AI for an intelligent digital twin: A case study on remaining useful life”. In: *Engineering Applications of Artificial Intelligence* 129 (2024), p. 107620. ISSN: 0952-1976. DOI: <https://doi.org/10.1016/j.engappai.2023.107620>. URL: <https://www.sciencedirect.com/science/article/pii/S0952197623018043>.
- [105] Alexander Lee. “Knowledge Representation and Reasoning in AI: Analyzing Different Approaches to Knowledge Representation and Reasoning in Artificial Intelligence Systems”. In: *Journal of Artificial Intelligence Research* 4.1 (2024), pp. 14–29.
- [106] Song Jie Lim, Varvara Vasilatou, and Shih Hsin Wu. “The use of CA to generate informal architectural systems”. In: *Proceedings of the 11th annual symposium on simulation for architecture and urban design*. 2020, pp. 1–8.
- [107] Chantel Hauptfleisch. “URBAN CELLULAR AUTOMATA AND AGENT BASED MODELS FOR THE SIMULATION OF URBAN DYNAMICS: A REVIEW OF PRACTICE AND APPLICATIONS”. In: (2020).
- [108] Francesco Orsi. “Centrally located yet close to nature: A prescriptive agent-based model for urban design”. In: *Computers, Environment and Urban Systems* 73 (2019), pp. 157–170.
- [109] Paulo Carreira, Vasco Amaral, and Hans Vangheluwe. “Multi-paradigm modelling for cyber-physical systems: foundations”. In: *Foundations of Multi-Paradigm Modelling for Cyber-Physical Systems* (2020), pp. 1–14.
- [110] Elias Lennevi et al. “Simulating Shazam Acoustic Fingerprinting for Music Identification”. report. Chalmers University, 2024.
- [111] Ryota Kusakabe et al. “Large-Scale Stabilized Multi-physics Earthquake Simulation for Digital Twin”. In: *Computational Science–ICCS 2021: 21st International Conference, Krakow, Poland, June 16–18, 2021, Proceedings, Part II* 21. Springer. 2021, pp. 3–15.
- [112] Spiros Mouzakitis et al. “Enabling maritime digitalization by extreme-scale analytics, ai and digital twins: the vesselai architecture”. In: *Proceedings of SAI Intelligent Systems Conference*. Springer. 2022, pp. 246–256.
- [113] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Towards an expressive, adaptive and resource aware network platform”. In: *Advances in Next Generation Services and Service Architectures* 14 (2011), p. 43.

Bibliography

- [114] Claudio Savaglio et al. “Generative Digital Twins: A Novel Approach in the IoT Edge-Cloud Continuum”. In: *IEEE Internet of Things Magazine* (2024), pp. 1–7. DOI: [10.1109/IOTM.001.2400035](https://doi.org/10.1109/IOTM.001.2400035).
- [115] Debashish Roy et al. “A complete overview of analytics techniques: descriptive, predictive, and prescriptive”. In: *Decision intelligence analytics and the implementation of strategic business management* (2022), pp. 15–30.
- [116] Shreya Sarker. “CIRCULAR ECONOMY Recycling Process Simulation Using OneM2M Communication Protocols: A Step Towards Digital Twins”. master’s thesis. Telecom SudParis and Asia Institute of Technology, 2024.
- [117] Luigi Liquori et al. “ETSI SmartM2M Technical Report 103716; oneM2M Discovery and Query solution (s) simulation and performance evaluation”. In: (2021).
- [118] Haya R Hasan et al. “Using NFTs for ownership management of digital twins and for proof of delivery of their physical assets”. In: *Future Generation Computer Systems* 146 (2023), pp. 1–17.
- [119] George Bruseker, Nicola Carboni, and Anaïs Guillem. “Cultural heritage data management: the role of formal ontology and CIDOC CRM”. In: *Heritage and archaeology in the digital age: acquisition, curation, and dissemination of spatial cultural heritage data* (2017), pp. 93–131.
- [120] Fabrice Dupuy et al. “The TINA-C Architecture as related to IN and TMN”. In: *Proceedings of the 3rd International Conference on Intelligence in Networks*. 1994.
- [121] P. Camarda et al. “A router for the interconnection of Ethernet local area networks via an ATM network”. In: *International Conference on Integrated Broadband Services and Networks, 1990*. 1990, pp. 283–288.
- [122] A Pelaggi et al. “From early ATM systems to advanced switching: issues in control architecture”. In: *CSELT TECHNICAL REPORTS 23* (1995), pp. 765–780.
- [123] Hendrik Berndt and Roberto Minerva. “Service Architecture”. In: *TINA Baseline Document TB 1600.012* (1995), pp. 2–0.
- [124] H Yagi et al. “TINA-C service components”. In: *Proc. TINA95 Integrating Telecommunications and Distributed Computing-from Concept to Reality* (1995).
- [125] Masaki Wakano et al. “TINA-C Service Architecture”. In: *Asia-Pacific Conference on Communications*. 1995, pp. 33–37.
- [126] Hendrick Berndt et al. *Tina Book: A Co-Operative Solution for a Competitive World*. 1998.
- [127] Roberto Minerva. “TINA service architecture: some issues in service control”. In: *Proceedings of the TINA*. Vol. 95. 1995.
- [128] CarloAlberto Licciardi et al. “Would you use TINA in your IN based Network? Some feasible scenarios”. In: *CSELT Technical Reports*. Vol. 25. CSELT, 1997, pp. 9–26.

- [129] C Capellman et al. "Migration Scenarios for evolving to TINA". In: *Proceedings of the TINA*. 1996.
- [130] CarloAlberto Licciardi, Roberto Minerva, and Alberto Cuda. "TINA is dead, long live TINA: toward programmable solutions for Next Generation Services". In: *TINA Conference*. Vol. 2000. 2000, pp. 16–20.
- [131] Roberto Minerva. "Middleware on the borderline between Internet and telecom services". In: *ICIN'98* (1998), pp. 66–71. URL: <https://cir.nii.ac.jp/crid/1571135649461834368>.
- [132] Roberto Minerva, Corrado Moiso, and Gabriele Viviani. "IL MIDDLEWARE COME FATTORE CHIAVE PER L'INTEGRAZIONE DI SERVIZI INTERNET E TELECOM". In: *CSELT technical Reports* 26.5 (1998), pp. 657–671.
- [133] Roberto Minerva. "Middleware on the borderline between Internet and telecom services". In: *ICIN'98* (1998), pp. 66–71.
- [134] Luigi Grossi et al. "New paradigm for network intelligence: From bundled to open and programmable systems". English. In: *Computer Standards and Interfaces* 21.2 (1999), p. 141. DOI: [10.1016/S0920-5489\(99\)92083-4](https://doi.org/10.1016/S0920-5489(99)92083-4).
- [135] Roberto Minerva and Corrado Moiso. "Will the circuits to packets revolution pave the way to the protocols to APIs revolution?" In: *CSELT TECHNICAL REPORTS* 28.2 (2000), pp. 213–226.
- [136] Roberto Minerva and Corrado Moiso. "Pushing Parlay one step beyond". In: *International conference on intelligence in next generation networks*. 2001.
- [137] Roberto Minerva and Corrado Moiso. "Next Generation Networks: should The New Service Architecture Approximate The IN?" In: *International Switching Symposium, XVII World Telecommunications Congress*. 2000.
- [138] Roberto Minerva. "Parlay/OSA: a reality check". In: *Parlay Meeting November*. 2003.
- [139] Roberto Minerva and Corrado Moiso. "The Death of Network Intelligence?" In: *International Symposium on Services and Local Access (ISSLS)*. Vol. 2004. 2004.
- [140] Mustafa Sabah Noori et al. "A performance evaluation of voice over ip protocols (sip and h. 323) in wireless network". In: *Proceedings of International Conference on Emerging Technologies and Intelligent Systems: ICETIS 2021 Volume 2*. Springer. 2022, pp. 381–390.
- [141] Roberto Minerva. "IMS as an enabler for Data Services". In: *4th International FOKUS IMS Workshop 2008*. Berlin, Germany, Nov. 2008. URL: <https://hal.science/hal-00630335>.
- [142] Roberto Minerva. "On some myths about network intelligence". In: *Proceedings of International Conference on Intelligence in Networks-ICIN2008 (October2008)*. 2008, pp. 1–6.

Bibliography

- [143] Roberto Minerva. “On the art of creating services: Do different paradigms lead to different services?” In: *Journal of Telecommunications Management* 1.1 (2008).
- [144] Roberto Minerva and Tiziana Demaria. “There is a Broker in the Net... Its name is Google”. In: *Preceeding of ICIN Conference*. 2007, pp. 1–6.
- [145] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “The Inner Circle: How to exploit autonomic overlays of virtual resources for creating service ecosystems”. In: *2010 14th International Conference on Intelligence in Next Generation Networks*. 2010, pp. 1–6. DOI: [10.1109/ICIN.2010.5640903](https://doi.org/10.1109/ICIN.2010.5640903).
- [146] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Bio-inspired autonomic structures: a middleware for telecommunications ecosystems”. In: *Autonomic Communication*. Springer, 2009, pp. 3–30.
- [147] Roberto Minerva, Antonio Manzalini, and Corrado Moiso. *Peer to peer adaptive networks to support Cloud Computing*.
- [148] Roberto Minerva. “How Adaptive Technologies can lead to a user centered Service Platform”. In: *SoftCOM*. invited speech. 2009.
- [149] Roberto Minerva, Antonio Manzalini, and Corrado Moiso. “Peer to peer adaptive networks to support Cloud Computing”. In: *CloudCamp Milan*. 2009.
- [150] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Autonomics in the home”. In: *2009 13th International Conference on Intelligence in Next Generation Networks*. 2009, pp. 1–6. DOI: [10.1109/ICIN.2009.5357074](https://doi.org/10.1109/ICIN.2009.5357074).
- [151] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Exploiting P2P solutions in telecommunication service delivery platforms”. In: *Handbook of Research on P2P and Grid Systems for Service-Oriented Computing: Models, Methodologies and Applications*. IGI Global, 2010, pp. 937–955.
- [152] Antonio Manzalini et al. *Service Composition Technology for the Internet of Services and for Next Generation Telecom Services*. 2010.
- [153] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “If the Web is the platform, then what is the SDP?” In: *2009 13th International Conference on Intelligence in Next Generation Networks*. 2009, pp. 1–6. DOI: [10.1109/ICIN.2009.5357108](https://doi.org/10.1109/ICIN.2009.5357108).
- [154] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Bio-inspired Autonomic Structures: a middleware for Telecommunications Ecosystems”. In: *Autonomic Communication*. Ed. by Athanasios V. Vasilakos et al. Boston, MA: Springer US, 2009, pp. 3–30. ISBN: 978-0-387-09753-4. DOI: [10.1007/978-0-387-09753-4_1](https://doi.org/10.1007/978-0-387-09753-4_1). URL: https://doi.org/10.1007/978-0-387-09753-4_1.
- [155] A. Manzalini et al. “Generalized Cognitive Networks of Virtual Resources”. In: *IX Workshop in G/MPLS (WGN9)*. 2010. URL: <http://hdl.handle.net/2117/11910>.

- [156] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Towards Resource-Aware Network of Networks”. In: *IEEE 5th International Symposium on Wireless Pervasive Computing 2010*. 2010, pp. 221–225. DOI: [10.1109/ISWPC.2010.5483787](https://doi.org/10.1109/ISWPC.2010.5483787).
- [157] Corrado Moiso et al. “Towards a service ecology for pervasive networked environments”. In: *2010 Future Network and Mobile Summit*. 2010, pp. 1–8.
- [158] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Autonomic clouds of components for self-managed service ecosystems.” In: *Journal of Telecommunications Management* 3.2 (2010).
- [159] A. Manzalini et al. “Self-optimized Cognitive Network of Networks”. In: *The Computer Journal* 54.2 (2011), pp. 189–196. DOI: [10.1093/comjnl/bxq032](https://doi.org/10.1093/comjnl/bxq032).
- [160] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Bio-Inspired Ecosystems for Telecommunications Pervasive Networks”. In: *Pervasive Computing* (2011), pp. 1–20.
- [161] Antonio Manzalini, Corrado Moiso, and Roberto Minerva. “Towards 0-touch networks”. In: *2011 Technical Symposium at ITU Telecom World (ITU WT)*. 2011, pp. 69–74.
- [162] Antonio Manzalini, Roberto Minerva, and Corrado Moiso. “Verso le reti 0-Touch”. In: *Notiziario Tecnico-Telecom Italia* 1 (2011), pp. 38–55.
- [163] Antonio Manzalini et al. “Specification, development, and verification of CAS-CADAS autonomic computing and networking toolkit”. In: *Formal and practical aspects of autonomic computing and networking: specification, development, and verification*. IGI Global, 2012, pp. 65–96.
- [164] Antonio Manzalini et al. “Autonomic Nature-Inspired Eco-systems”. In: *Transactions on Computational Science XV: Special Issue on Advances in Autonomic Computing: Formal Engineering Methods for Nature-Inspired Computing Systems*. Ed. by Marina L. Gavrilova, C. J. Kenneth Tan, and Cong-Vinh Phan. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 158–191. ISBN: 978-3-642-28525-7. DOI: [10.1007/978-3-642-28525-7_6](https://doi.org/10.1007/978-3-642-28525-7_6). URL: https://doi.org/10.1007/978-3-642-28525-7_6.
- [165] Antonio Manzalini et al. “Towards Halos Networks ubiquitous networking and computing at the edge”. In: *2012 16th International Conference on Intelligence in Next Generation Networks*. 2012, pp. 65–71. DOI: [10.1109/ICIN.2012.6376035](https://doi.org/10.1109/ICIN.2012.6376035).
- [166] Antonio Manzalini et al. “Software Networks at the Edge”. In: *Evolution of Cognitive Networks and Self-Adaptive Communication Systems*. IGI Global, 2013, pp. 199–226.
- [167] Antonio Manzalini et al. *Distributed Clouds at the Edge Networks*. 2013.
- [168] A. Manzalini et al. “Manifesto of edge ICT fabric”. In: *2013 17th International Conference on Intelligence in Next Generation Networks (ICIN)*. 2013, pp. 9–15. DOI: [10.1109/ICIN.2013.6670888](https://doi.org/10.1109/ICIN.2013.6670888).

Bibliography

- [169] Roberto Minerva, Antonio Manzalini, and Corrado Moiso. “Which way to QoS in future networks: Distributed or centralized decision logic ?” In: *2011 50th FITCE Congress - "ICT: Bridging an Ever Shifting Digital Divide"*. 2011, pp. 1–6. DOI: [10 . 1109/FITCE.2011.6133424](https://doi.org/10.1109/FITCE.2011.6133424).
- [170] Roberto Minerva et al. “Virtualizing Platforms”. In: *Evolution of Telecommunication Services: The Convergence of Telecom and Internet: Technologies and Ecosystems*. Ed. by Emmanuel Bertin, Noel Crespi, and Thomas Magedanz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 203–226. ISBN: 978-3-642-41569-2. DOI: [10 . 1007/978-3-642-41569-2_11](https://doi.org/10.1007/978-3-642-41569-2_11). URL: https://doi.org/10.1007/978-3-642-41569-2_11.
- [171] Roberto Minerva et al. “Virtualizing Network”. In: *Evolution of Telecommunication Services: The Convergence of Telecom and Internet: Technologies and Ecosystems*. Ed. by Emmanuel Bertin, Noel Crespi, and Thomas Magedanz. Berlin, Heidelberg: Springer Berlin Heidelberg, 2013, pp. 227–256. ISBN: 978-3-642-41569-2. DOI: [10 . 1007/978-3-642-41569-2_12](https://doi.org/10.1007/978-3-642-41569-2_12). URL: https://doi.org/10.1007/978-3-642-41569-2_12.
- [172] Serdar Vural et al. “Performance Measurements of Network Service Deployment on a Federated and Orchestrated Virtualisation Platform for 5G Experimentation”. In: *2018 IEEE Conference on Network Function Virtualization and Software Defined Networks (NFV-SDN)*. 2018, pp. 1–6. DOI: [10 . 1109/NFV-SDN.2018.8725755](https://doi.org/10.1109/NFV-SDN.2018.8725755).
- [173] Roberto Minerva. “Application Challenges for Operators”. In: *FUSECO FORUM 2010 on Future Seamless Communication*. Berlin, Germany, Oct. 2010. URL: <https://hal.science/hal-00630324>.
- [174] Pierpaolo Baglietto et al. “Analysis of design patterns for composite telco services”. In: *2010 14th International Conference on Intelligence in Next Generation Networks*. IEEE. 2010, pp. 1–6.
- [175] Roberto Minerva. “Trends in ICT Research Topics”. In: *Proceedings of ANDICOM CONFERENCE - COOPERATION DAYS EUROPEAN UNION - COLOMBIA*. CARTAGENA, Colombia, Oct. 2010. URL: <https://hal.science/hal-00630312>.
- [176] Roberto Minerva. “Looking back, looking ahead: Telecommunications and IT, a success story”. In: *ETIS Community Gathering 2011*. 2011.
- [177] Roberto Minerva. “Future platform technologies and architectures”. In: *ITU Workshop on "Service Delivery Platforms (SDP) for Telecommunication Ecosystems: from today's realities to requirements and challenges of the future"*. 2011.
- [178] Chunfeng Cui et al. “Network functions virtualisation”. In: *SDN and OpenFlow World Congress*. 2012.
- [179] Antonio Manzalini et al. “Clouds of virtual machines in edge networks”. In: *IEEE Communications Magazine* 51.7 (2013), pp. 63–70. DOI: [10 . 1109/MCOM.2013.6553679](https://doi.org/10.1109/MCOM.2013.6553679).

- [180] Katia Colucci and Roberto Minerva. “Una nuova filosofia per la ricerca: l’Open Innovation”. In: *Notiziario tecnico Telecom, Italia* (2008).
- [181] Roberto Minerva. “I paradossi della Rete: come sopravvivere in un Oceano Rosso e sognare un Oceano Blu”. In: *Notiziario Tecnico Telecom italia* (2009), Anno–18.
- [182] Roberto Minerva. “Telecommunications technology and .biz in this decade”. In: *Workshop On ICT For Innovation And Economy Recovery*. Murcia, Spain, 2010. URL: <https://hal.science/hal-00630317>.
- [183] Roberto Minerva. “Network Paradoxes and the Need to De-perimeterize Services”. In: *Second International Workshop on Business Models for Mobile Platforms (BMMP)*. <http://www.icin.biz/tutorials/2010#workshop1>. Berlin, Germany, Oct. 2010. URL: <https://hal.science/hal-00630319>.
- [184] Roberto Minerva and Steve Bell. *Boundary blurring between telecom and the Internet*. 2010.
- [185] Roberto Minerva. “Towards 2020: Some revolutionary paths for Telcos”. In: *2012 IEEE Technology Time Machine Symposium (TTM)*. 2012, pp. 1–1. DOI: [10.1109/TTM.2012.6509064](https://doi.org/10.1109/TTM.2012.6509064).
- [186] Roberto Minerva. “Disappearance of Telecommunications: the real biz starts now”. In: *COST-ICT Annual Progress Conference*. 2011.
- [187] Roberto Minerva. “Network paradoxes, personal data and the Future Internet”. In: *2nd ETSI Future Network Technologies Workshop*. Sophia Antipolis, France, Sept. 2011. URL: <https://hal.science/hal-00633772>.
- [188] Roberto Minerva. “Will the Telco survive to an ever changing world ? Technical considerations leading to disruptive scenarios”. Theses. Institut National des Télécommunications, 2013. URL: <https://theses.hal.science/tel-00917966>.
- [189] Roberto Minerva and Noel Crespi. *Networks and new services: a complete story*. Springer, 2017.
- [190] Jerome H Saltzer, David P Reed, and David D Clark. “End-to-end arguments in system design”. In: *ACM Transactions on Computer Systems (TOCS)* 2.4 (1984), pp. 277–288.
- [191] Marjory S. Blumenthal and David D. Clark. “Rethinking the design of the Internet: the end-to-end arguments vs. the brave new world”. In: *ACM Trans. Internet Technol.* 1.1 (2001), 70–109. ISSN: 1533-5399. DOI: [10.1145/383034.383037](https://doi.org/10.1145/383034.383037). URL: <https://doi.org/10.1145/383034.383037>.
- [192] Roberto Minerva. *SMART Cities: challenges and opportunities for communication operators*. 2011.
- [193] Antonio Manzalini, Roberto Minerva, and Vânia Gonçalves. “Halos Networks: A Competitive Way to Internet of-with Things”. In: *Communications and Strategies* 87 (2012), pp. 41–55.

Bibliography

- [194] Susanna Spinsante et al. “IoT-Enabled Smart Gas and Water Grids From Communication Protocols to Data Analysis”. In: *Internet of Things*. Chapman and Hall/CRC, 2017, pp. 273–302.
- [195] Roberto Minerva. *Internet of Things and the 5th Generation Mobile Network*. 2015.
- [196] Vincenzo Piuri and Roberto Minerva. *Building the internet of things*. 2015.
- [197] Roberto Minerva, Abyi Biru, and Domenico Rotondi. “Towards a definition of the Internet of Things (IoT)”. In: *IEEE Internet Initiative 1.1* (2015), pp. 1–86.
- [198] Yacine Ghamri-Doudane et al. “Guest editorial special issue on world forum on internet-of-things conference 2014”. In: *IEEE Internet of Things Journal 2.3* (2015), pp. 187–189.
- [199] Christos Verikoukis et al. “Internet of Things: Part 1 [Guest editorial]”. In: *IEEE Communications Magazine 54.12* (2016), pp. 12–13. DOI: [10.1109/MCOM.2016.7786103](https://doi.org/10.1109/MCOM.2016.7786103).
- [200] Christos Verikoukis et al. “Internet of Things: Part 2”. In: *IEEE Communications Magazine 55.2* (2017), pp. 114–115. DOI: [10.1109/MCOM.2017.7842420](https://doi.org/10.1109/MCOM.2017.7842420).
- [201] Christos Verikoukis et al. “Internet of Things: Part 3”. In: *IEEE Communications Magazine 55.3* (2017), pp. 108–109. DOI: [10.1109/MCOM.2017.7876966](https://doi.org/10.1109/MCOM.2017.7876966).
- [202] Christos Verikoukis et al. “Internet of Things: Part 4”. In: *IEEE Communications Magazine 55.9* (2017), pp. 14–15. DOI: [10.1109/MCOM.2017.8030478](https://doi.org/10.1109/MCOM.2017.8030478).
- [203] Yasir Saleem et al. “Parking recommender system privacy preservation through anonymization and differential privacy”. In: *Engineering Reports 3.2* (2021), e12297.
- [204] Yasir Saleem et al. “IoTRec: The IoT Recommender for Smart Parking System”. In: *IEEE Transactions on Emerging Topics in Computing 10.1* (2022), pp. 280–296. DOI: [10.1109/TETC.2020.3014722](https://doi.org/10.1109/TETC.2020.3014722).
- [205] R. Ayassi et al. “An Overview on Machine Learning-Based Solutions to Improve Lightpath QoT Estimation”. In: *2020 22nd International Conference on Transparent Optical Networks (ICTON)*. 2020, pp. 1–4. DOI: [10.1109/ICTON51198.2020.9203755](https://doi.org/10.1109/ICTON51198.2020.9203755).
- [206] Reda Ayassi et al. “Survey on the Use of Machine Learning for Quality of Transmission Estimation in Optical Transport Networks”. In: *Journal of Lightwave Technology 40.17* (2022), pp. 5803–5815. DOI: [10.1109/JLT.2022.3184178](https://doi.org/10.1109/JLT.2022.3184178).
- [207] R. Ayassi et al. “Field trial to assess Bayesian optimization for improving QoT estimation”. In: *49th European Conference on Optical Communications (ECOC 2023)*. Vol. 2023. 2023, pp. 389–392. DOI: [10.1049/icp.2023.2087](https://doi.org/10.1049/icp.2023.2087).
- [208] Alireza Farrokhi et al. “Application of Internet of Things and artificial intelligence for smart fitness: A survey”. In: *Computer Networks 189* (2021), p. 107859.

- [209] Praboda Rajapaksha et al. “Film Factory Losses: is BitTorrent a Major Responsible?” In: *2019 Sixth International Conference on Social Networks Analysis, Management and Security (SNAMS)*. 2019, pp. 181–188. DOI: [10.1109/SNAMS.2019.8931858](https://doi.org/10.1109/SNAMS.2019.8931858).
- [210] Dun Li et al. “A blockchain-based secure storage and access control scheme for supply chain finance”. In: *The Journal of Supercomputing* 79.1 (2023), pp. 109–138.
- [211] Wenhao Shao et al. “Video anomaly detection with NTCN-ML: A novel TCN for multi-instance learning”. In: *Pattern Recognition* 143 (2023), p. 109765. ISSN: 0031-3203. DOI: <https://doi.org/10.1016/j.patcog.2023.109765>. URL: <https://www.sciencedirect.com/science/article/pii/S0031320323004636>.
- [212] Dun Li et al. “Detecting market corners in crude oil futures: a robust hybrid anomaly detection approach”. In: *Applied Artificial Intelligence* (2024). URL: <https://hal.science/hal-04551522>.
- [213] Dun Li et al. “Blockchain-Enabled large language models for prognostics and health management framework in industrial internet of things”. In: *International Conference on Blockchain, Metaverse and Trustworthy Systems, BlockSys’ 2024*. 2024.

Bibliography

Part IV

Annex

RESEARCH BACKGROUND

A.1	Current Research Topics	84
A.2	More on the Modeling Problem	90
A.3	Current Prescriptive Templates	92
A.3.1	Aggregation of Cooperating Objects	92
A.3.2	Goal-based independent Objects operating in the same Environment	94
A.3.3	Open System Monitoring and Optimization	95
A.3.4	Scenario based Digital Twins	96
A.3.5	Multi-facet Modeling	100
A.4	Research over the Years	102
A.4.1	Preamble	102
A.4.2	Research Fields	102
A.4.3	Oh, That Book	107
A.4.4	Open Contributions	107
A.5	Additional Information	108
A.5.1	Teaching	108
A.5.2	A Bit of Metrics	108
A.5.3	Mentoring of Students	110
A.6	Acknowledgments	110

This Annex aims at complementing the research topics and activities discussed in the core document and framing them in a larger context of results achieved over the years. A first part is listing current research topics with some contextualization and explanation, the second part provides more information on the Modeling Problem, while the third sections gives more details about prescriptive templates as discussed in section 4.2.2. The Annex provides, in section A.4, an historical view on the research path. A fifth section presents a few metrics and additional information. Finally a section about acknowledgments to all the people that made this possible.

A.1 Current Research Topics

In this section a list of Research Topics is provided. There is not a particular organization of them, an attempt to list them according to priority was given, but it is difficult to prioritize them, so the criteria chosen is author's preference, first, and appearance in the document, second. They are referenced in the main part of the document within this convention RT.xx, where RT means Research Topic and xx is a sequential number, actually an identifier :) The Research Topics, RT, are:

1. **Digital Twin architecture.** This research points to the definition of a functional architecture capable of providing guidelines, rules and tools for enabling the rapid construction of well formed Digital Twin applications. It should support the implementation of DT's properties, the introduction of a data management and integrated layer supporting traditional sensing and actuation as well as the general purpose and synthetic sensing, a wide distribution of functions at the edge-cloud continuum, the re-composition of different models (prescriptive, descriptive, predictive with simulation), the integration with a DT-IDE, and the possibility to add plug and play functions related to AI techniques to extend the architecture's functionalities and modeling capabilities, and a set of APIs for programming services.
2. **Information Density .** It addresses the possibility through AI or other means to extract more information from available datasets. It is related to the possibility of further instrument an environment to collect more basic data to be transformed into information by algorithms or data fusion. On a more theoretical side, it is a measure of the inferrable level of information that the current available data sets can support with respect to the needs of applications. It also measure the need of further deploying sensors or collecting information for increasing the ability of available data to fully describe an environment or phenomenon. This problem can be related to general purpose and synthetic sensing: how much information can be extracted by the current general purpose sensing infrastructure? Should the sensing capability be increased? Is it possible to derive a theory for determining how much information can be derived from source data?
3. **Modeling Problem.** It addresses the interplay of different modeling capabilities needed to design, develop and execute a DT. It represents the consistent integration of Prescriptive, Descriptive, Predictive and Simulation Models (see other bullets point). At a general level, this problem points to the lack of methodology and tools for supporting the integration by maintaining the consistency. There is the need to develop mechanisms and functions to ease this difficult task relieving programmers to build from scratch the needed functionalities and the specialization of models. Due to the importance of this problem, the Reader is referred to section A.2.
4. **Bottom-up Modeling.** This research topic addresses the possibility of incrementally build a prescriptive model by observing and learning the behavior of a system by collecting data and derive rules, constraints, and normal or anomalous course of actions. This capability could be extremely relevant in the future of prescriptive modeling: a system could be described by a few characterizing properties and through analysis of data to infer its behavior. If specific representation objects are required, some Agents

(or plug and play functions) can be activated in order to extract the rules and constraints related to the specific objective. Construction of DT behavioral models could be greatly automatized if this research leads to positive results. This topic can find basis in Automata [90] [106], agent-based modeling [107], rule-based and case-based reasoning [108], [91], or multi-paradigm approach [109]

5. **Signature and DT** . This research topic points to the hypothesis of determining signatures that unambiguously identify objects operating in an environment. This capability can be used to determine the objects operating in an environment and progressively increase the information about them and the ability to characterize the object behavior. Different types of signatures can be determined as related to General Purpose Sensing capabilities. The wide adoption of Transformers and the possibility to represent data as charts/figures ensure the applicability of these techniques for improving the results. Madrid data analysis on correlation and redundancy widely used them [97] with good results.
6. **General Purpose Sensing applicability**. This research address the problem of understanding when and how general purpose sensing is a valid and viable alternative to specialized sensors networks. It also investigates what kind of sensing means are the most appropriate to collect the most of information.
7. **Synthetic Sensing Capabilities** . This topic investigates the limits and possibilities offered by AI and other transformation technique to extract/ infer valuable information from general purpose sensed data.
8. **Explainable DT**. This topic points to the need to explain why a DT is adopting certain behaviors with respect to different situations. This research will exploit the ability of prescriptive models to represent the behavior of an object and will associate it to the prediction and simulation capabilities of a DT to explain why the DT is following a specific course of action to cope with situations.
9. **Signatures Repository**. This research aims at verify the feasibility of a large repository collecting labeled signatures of as many as possible objects. These signatures represent distinctive marks of a classified object with respect to its impact on an environment. This repository could be used to map the newly acquired signature with existing labeled ones and to understand the type of object and its actions or impact on the environment. This repository could be seen as a large Shazam (the app for recognizing songs). Actually the technique adopted by Shazam, fingerprinting of noise [110], is a specific case of signature. The goal is to create a Repository to be able to collect different types of signatures characterizing objects and allowing their identification and classification in different contexts and environments. This Repository could have a great academic value (a collection of labeled signatures allowing the identification and classification of objects) and potentially industrial (e.g., the visual identification of malfunctions in engines or products).
10. **Situation Awareness for DT** . The Situation Awareness is an old concept that has not found yet a strong definition and wide spread applications. Some viable definitions are [40] and [41] that relate situation awareness to a context and its status. From

a very basic perspective, the DT definition needs to lay on a more detailed formulation. In this document, the concepts of environment, actors, context, and states of them are related to the understanding of what a situation is. A DT should correctly represent an environment and its actors. These elements should be contextualized according to the DT representation goals. In a context certain properties of the environment or actors are not relevant, while others acquire a great importance. The DT contextualization has a lot to do with goals, perspectives, and intended optimization of the DT behavior with respect to the "context". Think about the testing of a vehicle, in case of crash tests, the color, power, electric system, or other vehicle characteristics (represented by properties or values) are irrelevant with respect to the chassis design, the car shape and its speed. A situation should be represented according to relevant states (important for the context) in which the environment and its actors are. States could be normal, or approaching to critical values. A constant analysis of these state variables should occur during the DT execution. Situation Awareness means the ability to understand the global status of the PO with respect to a specific context of operation. The DT may also be able to take actions (Reactive or Autonomic DTs) to re-conduct the PO in a normal or optimized status. In an enterprise, a DT will be used by several groups with specific objectives and representation requisites (different views and different contexts). This creates further complexity and requires a clear segmentation and adaptation/flexibility, synchronization, and attribution of responsibilities to internal business processes.

11. **Simulation and Prescriptive models.** The DT's prescriptive model at execution time and its corresponding simulation model for a specialized simulation tool should be kept aligned: modifications made in the execution or simulation environment should readily be available in the other. In simple terms, the behavioral model should be unique and be interpreted or compiled for execution or simulation. This is a great challenge that is consequent to the definition of behavioral templates (that should be usable for execution and simulation)[49], [77].
12. **DT Composability** . A well formed DT platform should fully support the Composability property, i.e., the ability to create complex aggregations from simpler components or DTs guaranteeing the consistency of their modeling.
13. **Injection Chain.** The descriptive modeling of a DT is strongly based on the capability of feeding correctly (i.e., according to some formalization of data: data models or ontologies) the information needed for the DT representation and working. A correctly and effective injection chain is still a major requirement for feeding the DT. The chosen solution is FIWARE Data Models and the process to inject data into the Context Broker [15], [68], [71], [5].
14. **Prescriptive composition.** The coherence of prescriptive modeling is the ability to maintain consistency between prescriptive models even when specific environments or contexts of execution or particular subsystem behavioral aspects are considered in a DT and its components. The behavioral model of a component or DT at a low level of detail should be consistent with a general model (more abstract level, or a macro level) of the entire DT. A detailed behavioral model for elements of the complex physical object or a behavioral specialization of the DT to a particular environment

could be very useful and promising. The involved components should represent a detailed behavior within the special case, but provide a generalization of it when a more global perspective is adopted for the DT. For instance, simulations and tests about structural robustness of a vehicle's chassis could be considered as a detailed facet of the entire perspective on the vehicle. For crash testing, many particulars of the PO can be disregarded and the focus would be on fewer characteristics. However, these overlooked features and related behaviors should "re-appear" when a different facet of the DT is considered. This is related to multi-physics representations of DT [111] and extreme-scale computing applied to the behavior of DT[112]. Research is needed to achieve integration and support when prescriptive models at different levels are to be integrated to represent DT's behavior.

15. **Descriptive Composition.** A Physical Object can be observed from different perspectives. They may be extremely specific, but they are part of the whole description of the PO. There is the need to "Divide et Rule" the different perspectives maintaining global consistency of the data descriptions. A DT offering several facets can be represented by disparate ontologies, data model describing specific aspects of the physical object. The fusion or aggregation of these facets will provide a full understanding of the PO. For instance, the DT of a painting can comprise different facets focusing on the artwork as a physical object in need of particular conditions in terms of light, humidity and heat. At the same time, the object could be described in terms of materials and techniques used to "build" it. Another facet could be related to the artistic style and its interpretation over time. Another facet could be related to the intended usage of the object (a painting for a church, for a home, for exterior and the like). Other facets could be related to the history of the object (when it was built, under what circumstances, what happened to it along time and the like¹) or to the ownership of it (who was the committer, owners along time, is it a copy or the original and the like). Composibility property at the descriptive level, still, is a remarkable research issue in the DT definition.
16. **Behavioral Templates.** The availability of well-curated and proved templates could attract more developers with low expertise on DT modeling. In general, this approach aims at facilitating access to DT development resources, and setting up a library of general purpose components that can be tailored to the actual needs of a solution. This will avoid the burden of a long design and implementation time or the need for previous extensive experience in specification and development in this complex field.
17. **Scenario based DT .** This points to another type of DT capable of representing the consequences of changing variables in processes affecting a system. This activity includes the research related to the definition of Scenario (or planning) DTs capable of supporting a process definition and its simulation in different problem domains. The initial focus is on Smart City planning of recycling and traffic solutions. The goal is to find an optimal parametrization of entities and sub-processes needed to

¹The bust of Louis XIV of Bernini was made in Rome and was transported from Civitavecchia to Marseille by boat and then on a cart up to Versailles. This fact is full of implications like: why Bernini did not want to go to Versailles? How a statue of this value could be transported in those years? What were the commercial routes for a safe transportation? and many more.

complete an activity. This DT could also be supported by existing data that are used to calibrate the proposed optimization by means of DT simulation.

18. **FIWARE DT Architecture.** FIWARE is mainly oriented to descriptive modeling and it is very supportive to predictive models. The main point, for FIWARE and other platforms, is the lack of real support in the definition of a prescriptive model and, at run-time it is difficult to understand if the executing behavior of the Digital Twin is actually reflecting the operations of physical object. There is a need to much better support prescriptive modeling within these kind of platforms (FIWARE, but also others)[77].
19. **Architectural Expressive Power .** It points to the ability of a Software Service Architecture to provide suitable mechanisms, capabilities, and abstraction means to help in the design and implementation of Services exploiting the right level of simplification and abstraction [113]. This is particularly important with respect to Application Programming Interfaces Granularity.
20. **Network Digital Twin.** This points to the possibility of representing an Edge-Cloud complex infrastructure by means of a Digital Twin to exploit its capability of prediction of future states, simulation, and autonomic behavior for optimizing the resources usage. This NDT should support different roles and requirements of stakeholders by means of views on the allocated resources[63].
21. **Edge-Cloud Continuum.** It addresses the possibility of deploying DT components in a highly distributed fashion spanning over different administrative domains (segments). A DT or a distributed app can exploit the possibility offered by deployment over several segments with different features. This comprises also an Edge-Cloud optimization [76]. It is a more general problem that can be addressed by considering segments with different characteristics in terms of computing, communication and storage (and also sensing).
22. **Multi-Facet DT description .** It is the research issue related to the different perspectives that can be used to represent and observe the behavior of a complex object. The current trend is to describe the object in a comprehensive Ontology that can grow and becoming very complex because different aspects of an object are to be represented. On the behavioral side, the object can have distinguishable behavior depending on the environment and intended usage. This research aims at identify different descriptive and functional/behavioral facets of the complex object and to guarantee their integration. The object, when needed, can be analyzed form a specific facet still maintaining its coherence with the overall definition and structure of the system. This research is related to behavioral and descriptive composition already mentioned.
23. **Integrated Development Environment for DTs .** This topic points to the need of a well and complete Integrated Development Environment capable of associating Descriptive, Prescriptive, Predictive and Simulation Modeling in such a way to ease the construction, modification and testing of DT solution. It should integrate different existing components and methodologies, such as MBSE, and supporting top-down,

as well as bottom-up modeling capabilities. It should also guarantee the consistency of prescriptive model during simulation.

24. **Trajectory Prediction in Urban Environments** . This research topic points to the improvement or definition of techniques for determining the most likely trajectories of vehicles in urban contexts. The work of [93] has been further consolidated in [114].
25. **General Purpose and Synthetic Sensing Federated Learning** . This research addresses the way to activate different threads of data collection and transformation by means of general purpose and synthetic sensing aiming at monitoring very local phenomena and trying to reconnect them on a more global scale. The monitored phenomena can be of the same type (e.g., traffic intensity detection), in this case the Federated Learning has the goal to provide evidence about the peculiarities or similarity of a phenomenon in different locations. The data transformation could also be applied to different phenomena related to a common general problem: for instance certain edge nodes could use video for calculating traffic intensity, while others will use noise or pollution. In this case the re-composition and understanding of generated data will be a major goal of this federated learning ability.
26. **Wide AI usage for GPS data transformation** . This topic explores the automatic ability of the GPS system to train and deploy at the edge different algorithms with parametrizable variable to identify and classify information, signatures and contextual data to be successively transformed into information. A couple of interesting approaches may be attempted: the opportunistic DT could start trying a set of pre-defined AI algorithms and progressively converge towards those that provide the best correspondence with the intended DT's goals; on a more extreme heuristic approach, the set of test algorithms could be increased along time, looking for those that transform GPS sensed data in some environment information. In a way, a "try and error" strategy could be considered to understand what information to extract from the environment by means of GPS. This is a consequence of moving the complexity and specialization from the hardware infrastructure to the software layer.
27. **Sensor placing Methodology** . This points to the possibility of using local and temporary general purpose sensing devices to measure the correlation and similarity of candidate locations with existing sensors. The best location to deploy is based on the wanted deployment density (number of sensors) but also on the minimal redundancy, i.e., a lower correlation between sensors. GPS devices or mobile ones could be used to progressively finding low correlated positions according to the grid of deployable sensors. The goal of this activity is to find a trade-off between large deployment with high level of redundancy and correlation, and smaller ones capable of not losing relevant information in local areas.

Other Research Topics may have been presented and not comparing in the list. However, almost 30 research topics seem to be enough even for large and curiosity eager research groups.

A.2 More on the Modeling Problem

The Digital Twin is a contextualized model representing the states and actions of a system (RT3). The modeling is a major issue for the success of a DT implementation. In the industry, best practices for a descriptive and a prescriptive modeling is consolidated [13]. However, the interpretation of these modeling requirements and the applicable techniques can be divergent. Here, the descriptive model represents the status of a system that exposes the "value" of its attributes with respect to the context under analysis; while the prescriptive model represents actions and actuated behavior by the system to move from one status to the other in a specific context (and producing new values for the states). In a way, prescriptive model represents the actions, behaviors of the system under consideration with respect to a specific context, while the descriptive model represents the effects of that behavior.

In addition, the enhanced capabilities of Artificial Intelligence can help to predict future states of a system based on historical data. This introduces another dimension to the modeling: the predictive model, creating a triad of modeling capabilities [115]. Besides, one of the major properties of the Digital Twin is simulation. This can be seen as a projection of the prescriptive modeling into a different time (in the future for understanding potential behaviors, or into the past to analyze why and how the object had a certain behavior). A system can be observed as a succession of states over time without considering what

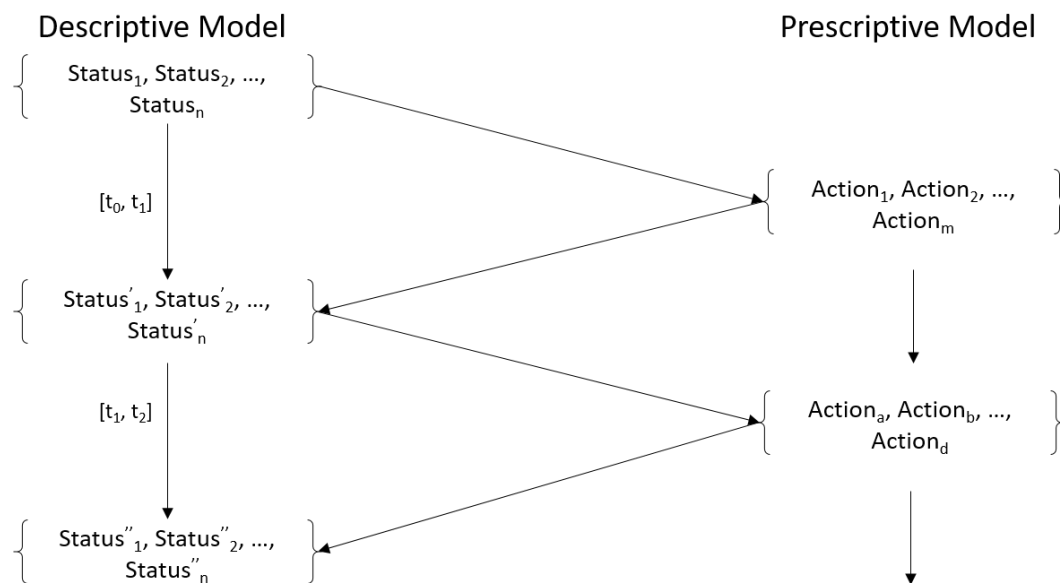


Figure A.1: Interplay of descriptive and prescriptive Models

actions were needed to change the states as suggested by Fig.A.1.

In a descriptive model, the succession of states describes the steps that led the system into current state. No justification about the reasons for this course of actions can be directly derived. A prescriptive model works on data and exerts actions so that it can transition to a different state (hence, transforming the data representing status of the system). The prescriptive model and related actions correspond to a DT implementation.

Fig.A.1 represents the interplay between the initial status and actions carried out on data over a succession of time steps and the related changes of states of the system under obser-

vation. The combination of prescriptive and descriptive models provides a full understanding of the system’s behavior and the analysis of transitions can also explain the reasons behind certain actions. A prescriptive model is necessary if the system needs to be fully represented and understood (RT3).

System’s future states can also be predicted by Artificial Intelligence, e.g., ML / DL, mechanisms (see Fig.A.2a). They can provide more or less accurate results (i.e., predictions on states changes) depending on the characteristics and complexity of the system, but, reasonably, if the historical data are consistent and cover different situations, the predictions will be factual.

Fig.A.2a depicts the possibility of using historical data to predict with accuracy the future status of the System and DT. Obviously, in this case, the typical opacity of Machine / Deep Learning techniques in explaining reasons for specific results should be taken into account. Another possibility offered by a well designed Digital Twin is Simulation of the

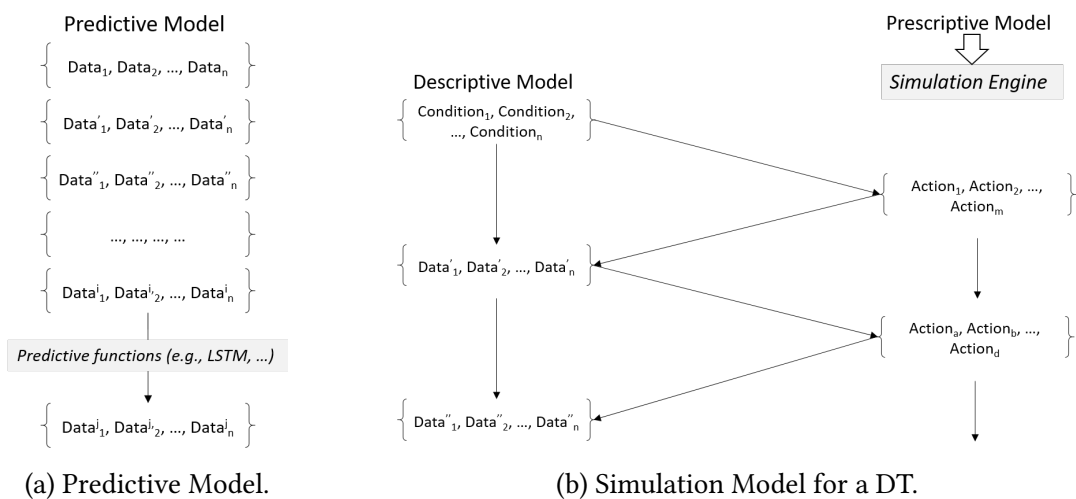


Figure A.2: Interplay of different Modeling.

physical system behavior. Fig.A.2b represents the case in which the system is placed in an initial condition (different from the usual ones) and the prescriptive model is used to reproduce the expected behavior of the system and generate status data that represent the simulated actions. In this case, the prescriptive model can help in understanding why the DT is reacting to certain conditions and what the resulting behavior and transition to states could be. One important factor to consider is that the simulated behavior and the implemented one should be exactly the same, otherwise there could be discrepancies between simulation results and those occurring in reality (RT.11).

These examples point to the need to integrate the different DT modeling needs in order to provide a usefull and usable representation of the complex physical object capable of contextualizing it according to needs, but still providing actionable insights. The modeling is a major issue of the DT representation and it should be better supported in current development systems. Actually, even the best technologies [16] do not support the full implementation and the alignment of different models (prescriptive - descriptive, prescriptive - simulation, prediction - prescription). This topic is extremely important and should also

be considered in conjunction with the lifecycle modeling requirements. Well formed templates of DTs types would be important for guaranteeing the alignment and for easing the activities of developers.

A.3 Current Prescriptive Templates

In order to align the different modeling requirements and for better support different way for describing the modeling (e.g., top-down or bottom-up) a set of behavioral templates have been identified to proceed with the research about RT.16, RT.4, RT.23, and in general RT.1. An initial set of prescriptive templates has been identified and is under consolidation [77]. They are listed here:

- A. Aggregation of cooperating objects with goals in a single system;
- B. Goal-based independent objects operating in the same environment;
- C. Open system monitoring and optimization;
- D. Aggregation of processes and subprocesses.

Other templates will be progressively identified and specified to create a library of components that can be specialized and improved within the Telecom SudParis' DICE Lab and, in the future, proposed to a larger audience.

A.3.1 Aggregation of Cooperating Objects

Fig.A.3 represents a oil vessel fully instrumented with sensors and actuators. The goal is to measure liquids' levels and pressures in such a way to regulate the incoming and outgoing flows, and separate different fluids. To achieve this goal, there is a need to open several valves to maintain a safe operational level and effectively separate the different elements. The "situation" concept is instrumental to the behavior definition. A situation can be described as the set of systems parameters/attributes that represent the actual status, in terms of states of system's components and environment. If some of them approach or exceed critical thresholds as a consequence of actions or events, the situation can be termed critical and these parameters need to be re-conducted to normal values. During normal operation, the DT can operate on a well defined set of states, if critical events do occur, the set of parameters to consider for representing the situation may change (i.e., some may have priority over others). A expected behavior is represented on the right part of the Fig.A.3. In this case, it is easy to identify basic actions and general constraints. When a certain situation occurs, then a set of basic actions can be executed. In case of critical situations or events, different course of actions should be taken.

The consolidation of these ideas is going through a more formal specification using MBSE supported by the eclipse's Papyrus tool². Fig.A.4 represents a formalization of the previous vessel example by means of the Papyrus tool. Ideally an Integrated Development Environment, IDE, devoted to DT specification and implementation could provide access to the templates and to a vocabulary of well formed "basic actions". The developers could use them to set the scenario of usage and then to specialize the behavior by

²<https://eclipse.dev/papyrus/>

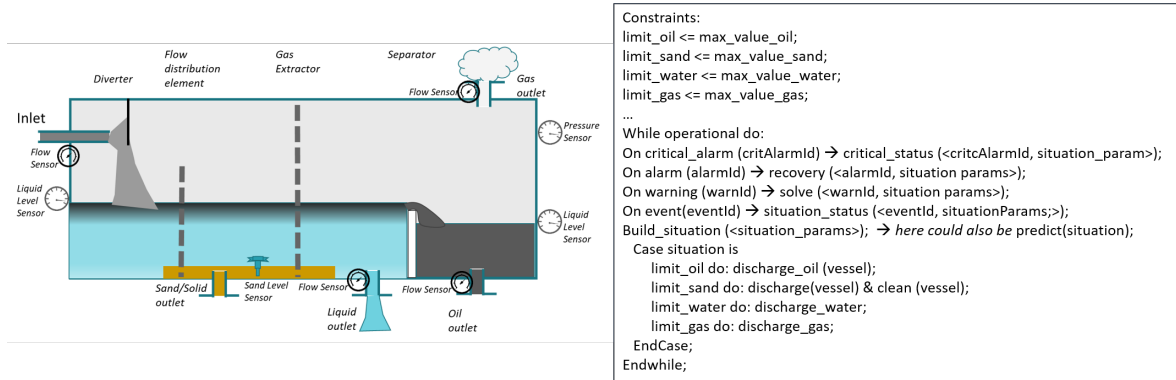


Figure A.3: Aggregation of Objects with a common Goal

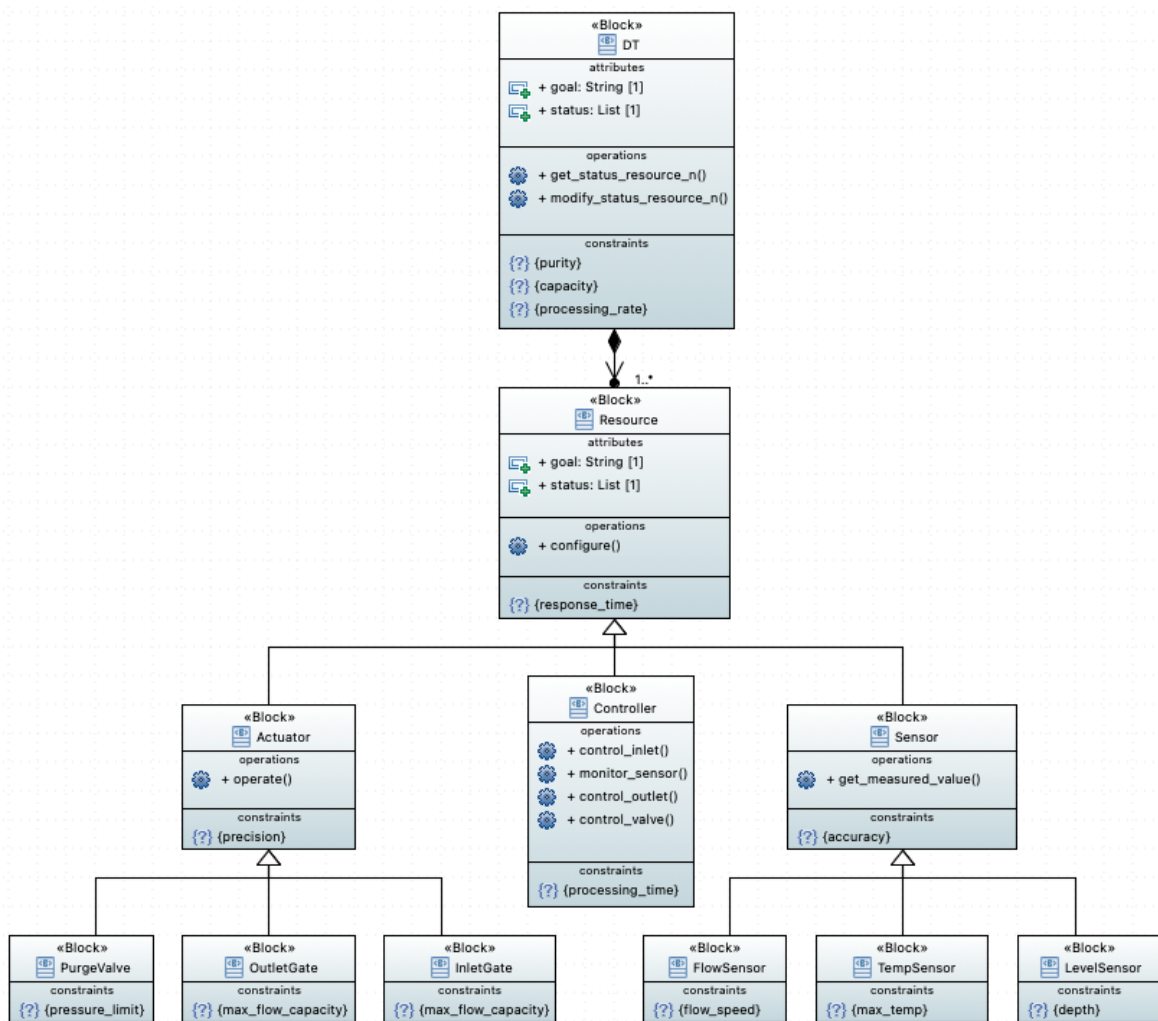


Figure A.4: Representation of an Aggregation of Objects with a common Goal

changing/introducing new goals, basic actions, parameters of relevance and constraints. In addition, data could be injected in the DT system and compared to those expected in order to calibrate the prescriptive model of the DT with actual actions. In this case, the modeling of the environment is not needed because the system integrates them into its representa-

tion (e.g., the temperature of the external environment can impact the internal one of the vessel, but sensors in the tank are providing the needed data to check, intervene, control and govern the system).

A.3.2 Goal-based independent Objects operating in the same Environment

Another basic template is the "Goal-based independent objects operating in the same environment". In this case, a set of independent objects are operating in a common space that has obstacles, off-limits areas, but also shareable resources (to allocate, use and release), and generic "posts" in which to take, or leave objects. An exemplification of this could be a factory with independent robots operating in the same production environment. The cooperating entities need to negotiate for space, resources and objects when they arrive close to them and close each other. Negotiation on priority can be defined based on the entity needs and its task "priority" within a global process. Fig.A.5 represents two objects with intersecting tasks in the environment. Scripts on the left represents a fragments of their simplified behavior. The explicit introduction of an environment representation is extremely useful

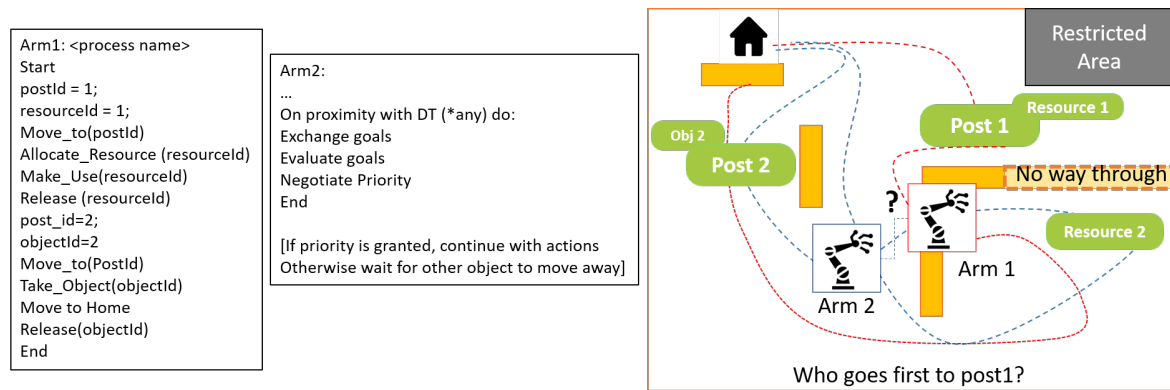


Figure A.5: Representation of Cooperating Objects within a Common Environment

because it allows to specify rules of behavior as well as obstacles, and constraints (such as off-limits zones). Also resources and common areas of management of objects can be defined. In addition, the cooperating entities may allocate and deallocate resources, but also they can negotiate between them for prioritizing their activities in case of interacting actions or requests for access to same areas or resources.

A more formalized template for this scenario is provided in Fig.A.6. This template could be modified and reused for different scenarios such as warehouse management, or others. In principle, it can be reused with extensions and modifications for representing traffic in smart cities.

An extension of it (still to be fully formalized) is "Goal-based independent objects operating in different environments with interconnecting spaces": for example two factories working on different objects to be assembled. Sub-products are to be shipped by means of some communications means (that are interconnecting spaces) when ready. Communication between objects in the different environments can be needed and supported by the template to synchronize the actions and passage through interconnecting spaces. Another example is MapReduce processing of a distributed system. Different computations in far

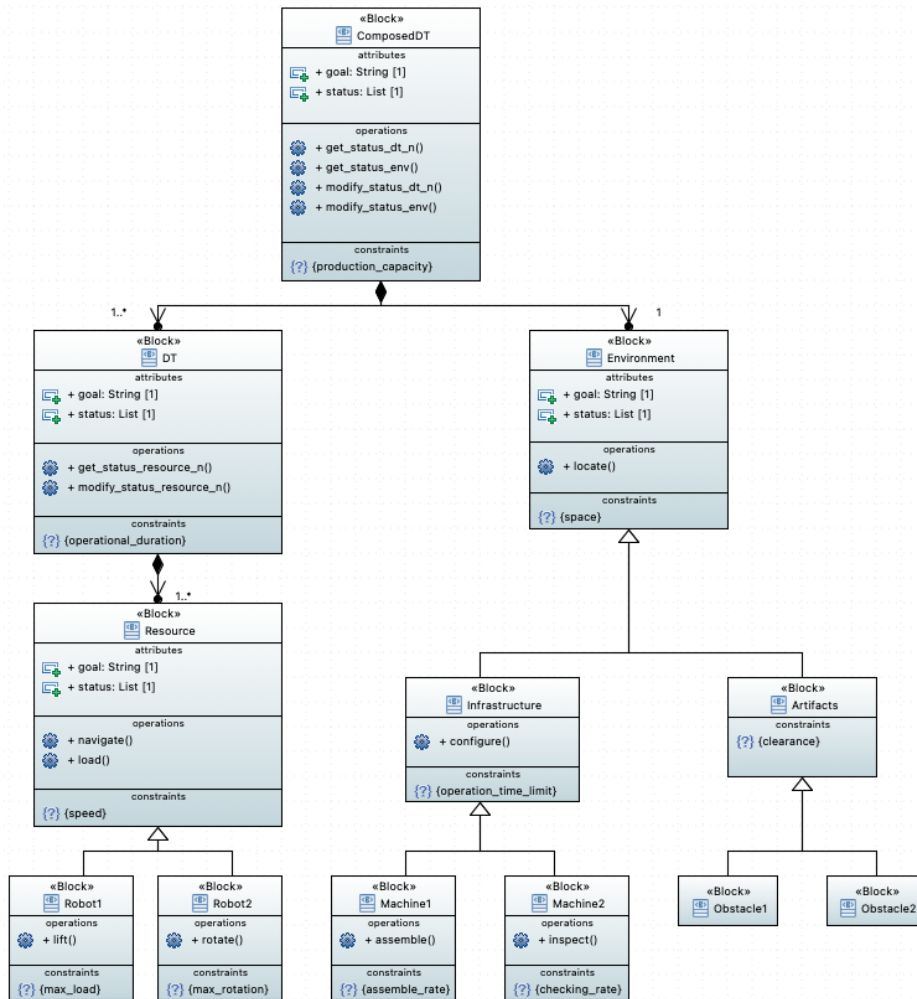


Figure A.6: Towards a formal Representation of cooperating Objects with Papyrus

away segments can take place for the Map function, when finished, they communicate the results to interconnecting spaces that will send data to specialized objects for recomposing / reducing the results (a DT supported Hadoop model).

A.3.3 Open System Monitoring and Optimization

Another identified template is related to a DT devoted to monitoring of a system and its optimization (local, global and stakeholders' related). In this case, the environment comprises several resources, placed in distributed locations and possibly owned by different owners. The environment poses and represents stringent allocation rules and constraints on how to use the available resources. The entire system needs to be optimized in two ways: minimization of used resources, and maximization of allocated resources' usage from the stakeholders views. In this sense, the DT has a global optimization goal and a set of stakeholders goals. The global one can be specialized in sub goals of the entire system and sub-goals of individual resources. The environment has a global status that reflects the states of resources and it has a behavior. It has an owner, responsible towards stakeholders of the allocation and usage of resources.

Resources are allocated on-demand or according to policies (e.g., Service Level Agree-

Annex: Research Background

ments, SLAs). Resources emit events (completion of actions), warning (approaching to a threshold), alarms and critical conditions. Resources have location, global capacity, owners and can be allocated to users according to global or local policies that are compliant to SLAs. Basic actions can be performed on resources (e.g., typical management actions). Actions are instrumental to achieve goals. Actions can be collected into tasks to perform policies with a relevant goal.

Stakeholders are users of the environment and they have defined contracts with related SLAs. They allocate resources for their purposes and associate them to their Entities (e.g., applications, policies, monitoring) for the execution of internal (and not necessarily visible) tasks. They execute actions (by means of their entities) on the system. Fig.A.7 represents a

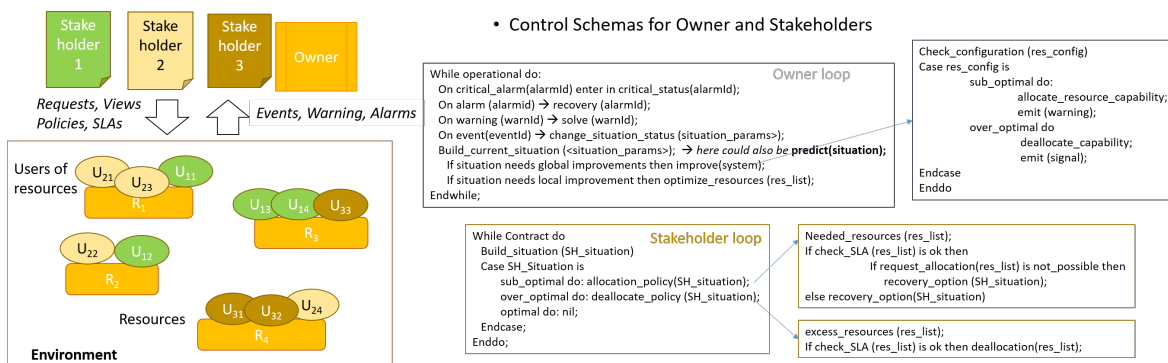


Figure A.7: Representation of a monitoring System

simple case of resources allocated to different stakeholders and their users and two monitoring loops to optimize the system from a global view of the Owner and maximization of resource usage from the stakeholder standpoint.

Some possible application scenarios are Network Digital Twin or large Data center devoted to Disaster Recovery Systems.

A.3.4 Scenario based Digital Twins

Another way of using a Digital Twin is for planning processes and actions in a specific environment (RT.17). The European project Climaborough³ is working towards the identification of critical issues for helping smart cities to correctly cope with climate problems by means of practical solutions capable of contributing to achieve a better CO_2 footprint. In this context, the DT approach is considered with interest under two different perspectives:

- Predictive Digital Twin, based on available data it will predict the evolution of a phenomena in the city and will also provide predictions on the CO_2 footprint;
- Planning or Scenario based DT, it will describe a process within a city and will help planning needed resources and activities to carry it out in an optimized way (minimization of activities, CO_2 footprint, and costs associated to the processes).

Fig.A.8 represents two types of DT considered in the Climaborough Project [78] and some of their differences. The scenario based DT will work on the description and representation

³<https://climaborough.eu>

of processes, on some relationships and constraints on available resources, the concatenation of sub processes, and on the definition of functions to be optimized according to goals and constraints posed by Users. Fig.A.9 represents the Climaborough’s platform function-

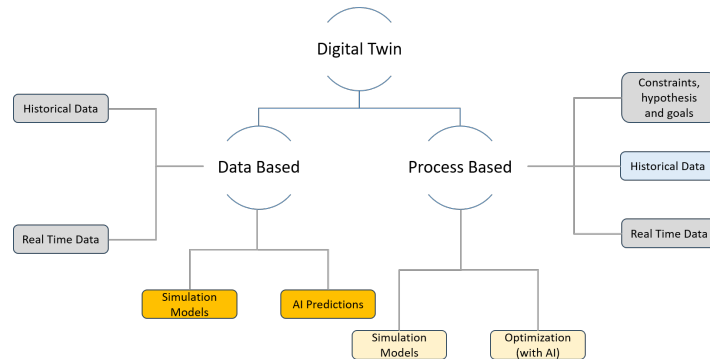


Figure A.8: Climaborough Project: Two types of Digital Twin for the Smart City

alities devoted to the deployment and execution of relevant DTs. In this section, we focus

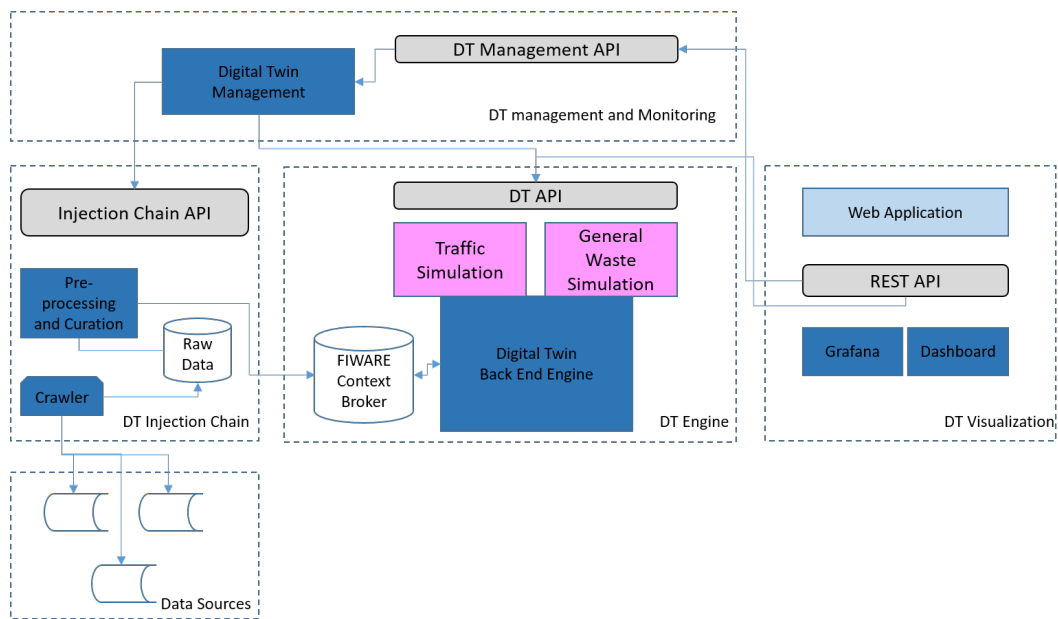


Figure A.9: Climaborough Project: generic Architecture for supporting different DTs

on the Scenario (or Process based) DT because it poses relevant challenges from a modeling perspective. The idea behind this type of DT model is to represent complex processes in terms of entities and resources involved and to determine what are the possible outcomes of these activities. In the case of a recycling process in a smart city, many entities and resources are involved. As a simplified example, Fig.A.10 represents a set of recycling activities related to Entities involved. Intelligent Bins (i.e., bins with sensors and edge computing capabilities are assumed) are an essential element of the process (and one of the main resource to optimize). Then there are trucks for collecting material, a coordination Central Control and (not shown in the picture), one or more Warehouses where to store and recycle the material. The Central Control is in charge for the coordination of activities

Annex: Research Background

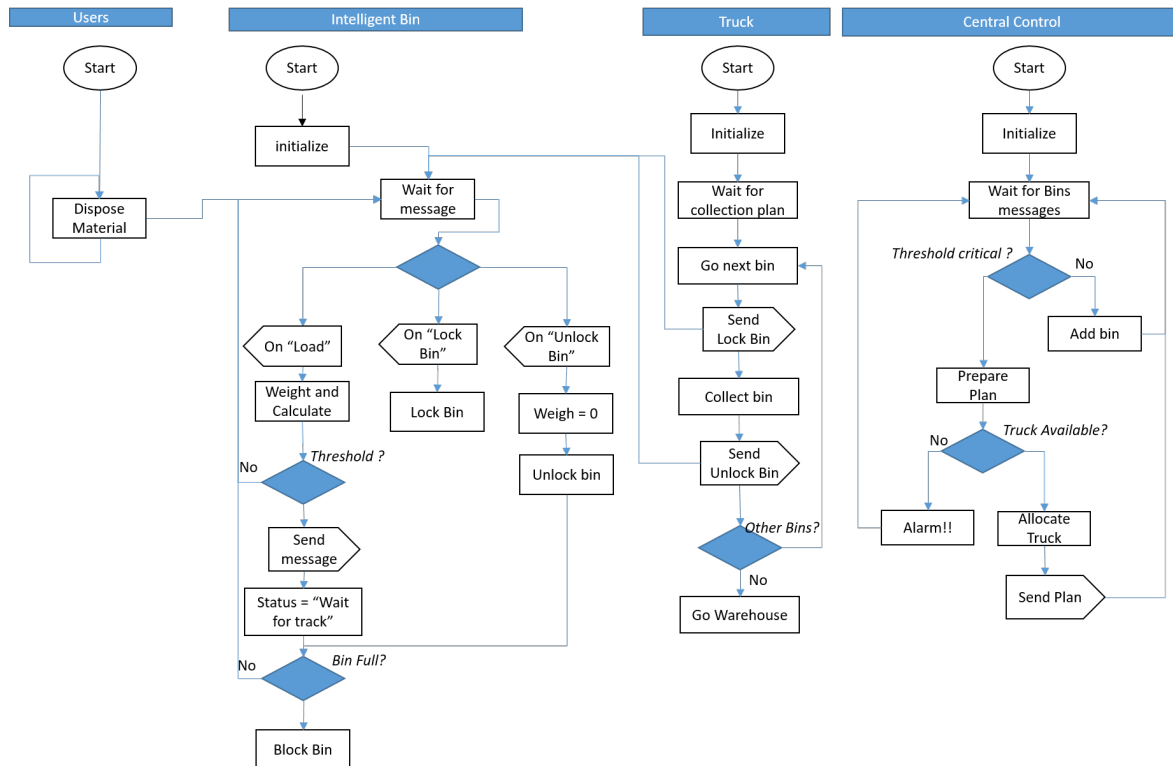


Figure A.10: A simplified recycling Process

by considering the status of all available resources. In order to implement effective policies, resources can communicate. In fact, some relevant communicating entities are depicted. They communicate each other by means of messages (as consequences of events occurred locally). A bin collects the disposed materials by users. When a threshold is reached, a request to collect is send to the Central Control. This entity is keeping track of bins' levels as well of the available trucks. It prepares plans for collection trying to optimize the truck usage, the level of bins and the total garbage collected by trip. It also considers the capabilities of Warehouses to avoid overwhelming their recycling capabilities. There is a need to optimize the bin collection process (when should they emit requests for collection), tracks availability and their trips (reducing trip length), the warehouse(s) capability to storing and recycling the material. There are different local optimizations and they need to be coordinated to achieve an effective global optimization. Knowing the total and actual capacity of each bin and truck is important to design an efficient track collection trip without leaving bins uncollected. On the other side, the capacity of warehouse needs to be aligned with those of trucks and with its ability to transform the disposed material. Trucks should not collect more disposal than the total quantity allowed by warehouse(s); on the other side, full bins should not be left uncollected. Too many trips of trucks are to be avoided as well as to collect too often almost empty bins. On a global perspective, there are at least two other goals to reach: minimize the carbon footprint of the entire process as well the total costs associated to these processes. Fig.A.11 represents three different functions that need to be optimized with respect to this problem domain. They can be supported (if existing) by real-time data acquired by the resources on field. A DT could be defined to optimize these functions according to the input parameters provided: e.g., number and features of

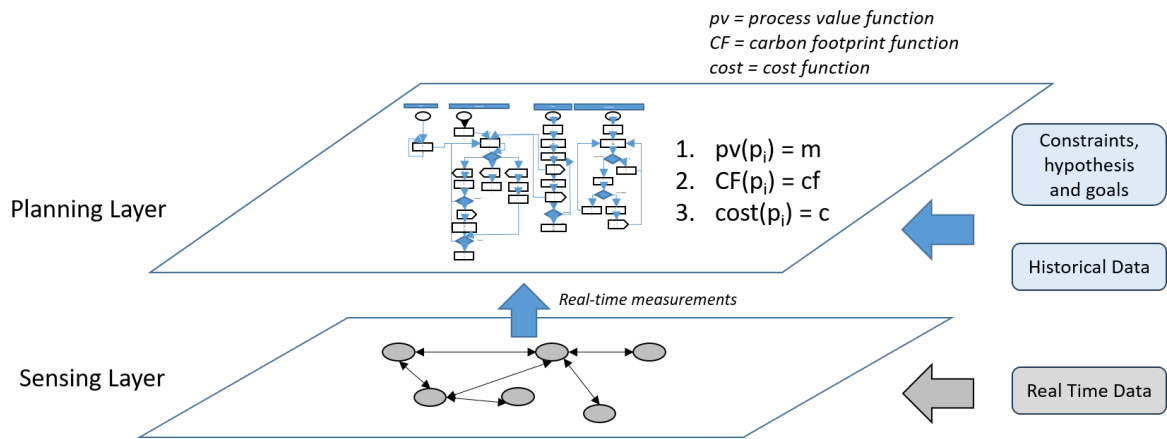


Figure A.11: Problem Functions and possible Relations to actual Data

available bins (their location, capacity, expected rate of disposal by users and other features); number and characteristics of trucks (their type, capacity, consumption, and type of vehicle, maintenance requirements and more); warehouses features (e.g., storage capacity, transformation rate and timing, consumption and Carbon Footprint, and others). If these features are known, a DT capable of optimizing these functions can be implemented and executed. This optimization will act upon the available parameters and will improve the three goal functions:

1. The ProcessValue function measures the total amount of possible collected material. Hypothesis: the whole process will be divided in sub-processes, $PV = \{p_1, p_2, \dots, p_n\}$. Each sub process p_i will produce a quantity m_i . All the material that can be recycled by the global process is the result of the sum of all the collection supported by sub-processes:

$$PV = \sum_{i=1}^n f_i(p_i) = m \quad (\text{A.1})$$

where m is the total recycled material. Each sub-function will have parameters and variables, they are related to quantities to be calculated; parameters are values that can influence the quantity in a specific sub-process (e.g., the density of recycle bins for km/square, the capacity of each single bin, etc.). Identification of relevant parameters will be fundamental for designing “good function” and sub-functions $f_i(p_i) = (a_1 + a_2) x_1 + a_3 x_2 \dots$

2. The Carbon Footprint function:

$$CF = \sum_{i=1}^n cf(p_i) \quad (\text{A.2})$$

It will measure the carbon footprint expected by the whole process as a sum of all sub-values for sub-processes.

3. The Cost function will calculate the sum of costs associated to each p_i sub-process:

$$C = \sum_{i=1}^n c(p_i) \quad (\text{A.3})$$

The PV function will have a leading role: when it will find a small set of parameters that represents acceptable or good balance between usage of resources avoiding overwhelming of bins, trucks, and/or warehouse(s), the other functions will play a role in determining the best solutions. Depending on the choices of users, cost benefits or lower carbon footprint could be prioritized. The outcome of DT modeling is the optimization of given resources and associated costs. In practical terms, the DT will determine the best way for organizing the resources to get the maximum possible quantity of recycled material, reducing the Carbon Footprint and Costs.

There is another way of using this type of DT: as a sort of reverse engineering tool. Given the expected total recycled material (it is the general goal), to determine the minimum number of resources needed to achieve the target optimizing as well Carbon Footprint and Costs. Actually, this way of working for a DT is extremely useful for planning and investigating the impacts of new solutions. An important problem arises: data supporting the hypothesis are not necessarily all available (e.g., the number and capacity of smart bins), so there is a need to deal with variables and uncertainty. However, it is also possible to use another important capability of the DT: simulation. Defining a template of the processes and using parametrization of their salient characteristics can result in a means to find acceptable and optimal solutions. A Master's thesis [116] has laid the basis for a "general model template" for recycling activities and to support it with a simulation engine (based on OMNET++ tool and ETSI's oneM2M specifications [117]). The processes definition will be further improved by means of co-design tasks with representatives of Smart Cities in the course of Climaborough Project's activities. The goals of this research is then:

- **Direct Calculation**, i.e., identification of sub-processes, characterizing parameters (e.g., density of the bins per km/sq, bin size, number of trucks, their size and capacity, volume and capacity of warehouses, their number and locations, ...) and calculation of the total recycled material upfront to a measurement of total Carbon Footprint and associated costs;
- **Reverse or Goal calculation**, i.e., given a goal (e.g., target amount of recycled textile) to refine the processes, identify a set of values for the parameters that can satisfy the goal and provide relevant indication about the needed resources for achieving the expected results (comprising Carbon Footprint, and Costs).

The specification and simulation of processes for recycling materials will be instrumental to a sound definition of a template for Scenario based DT. It will be consolidated and improved with the help of additional stakeholders and related demos.

A.3.5 Multi-facet Modeling

This section is addressing the application of a DT concept to Cultural Heritage [30], but it also presents a research topic of general value (RT.22): the integration of different ontologies and/or data models for describing the complexity of an object from different perspective. In this case, the physical object is an artwork. Obviously, a DT can represent an object of art. This representation opens up different technical and business perspectives. For instance an artwork can be associated to a Non Fungible Token, NFT [118] and the different replicas can be "sold" to people. In this domain, a full DT representation needs to comprise different views that describe the many aspects/facts of the artwork.

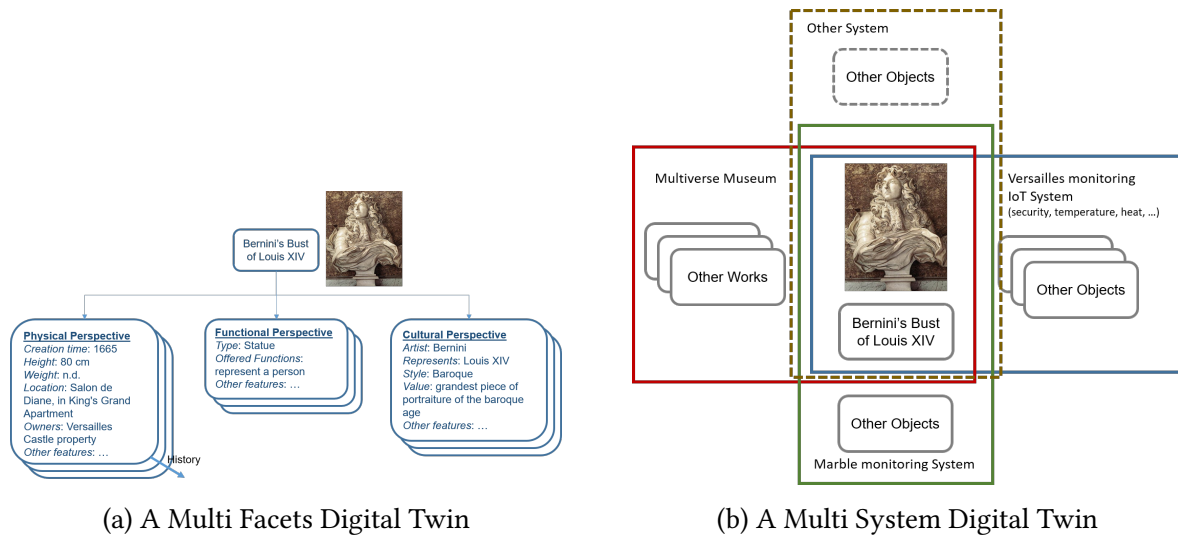


Figure A.12: A DT Representation Of Bernini Bust of Louis XIV showing different Facets of its Representation and its possible Integration into different Systems

Fig.A.12 represents how an object may be described and modeled from different perspectives and the same object can also be integrated in different systems (and related DTs). Typically, ontologies and data models created for Cultural Heritage tends to be complicated because they address different perspective into a single model (see for instance CIDOC CRM [119], one of the most important ontologies in this field). Some consequences of this monolithic approach are an increasing complexity and rigidity of the representation and, consequently, the difficulty to separate concerns. The DT should instead support different perspectives on the same system/object and should allow for fast representation and description of the behavior from a specific standpoint. On a system side, the same DT should be easily integrated in a different environment / system and contribute to its operation according to specific concerns and needs. These hypothesis are currently under study and have not found yet their way towards publications. However they are considered relevant hypothesis to corroborate with experiments and models.

From a descriptive modeling, the multi facets could have a positive result: different integrated data models (RT.15) referring to the same object can contribute to separation of concerns and optimization of descriptions with the capability of detailing aspects of the object according to a specific perspective. On a prescriptive side, the same object can be utilized in several manners with behaviors description (RT.14) aligned to the specific view. A detailed model may be needed to understand the behavior from a specific perspective (e.g., the material composition of an artwork). On a more general view, the detailed behavior should be "summarized" (but should be kept consistent with the specialized model) to represent the global behavior (e.g., the reaction to humidity of a painting). On a way, this approach could be a simplification, but the need for alignment of internal state of the object operating according to proper models in global or specialized environments is a great research challenge.

On the prescriptive side, multi facets are very much associated to the concept of multi-physics: different subsystems of a large entity can be modeled according to specific physics

laws (for instance, subsystems of an aircraft can be supported by different branches of physics, mathematics and heuristics: aerodynamics, communications, mechanics, material science, and more). All these different representations need to be integrated and work well together to correctly represent the system status. In the large, descriptive and prescriptive models should be integrated at different levels of detail to help in correctly represent the salient features of a large system. This is currently achieved in specific and proprietary solutions. There is a research need to clarify and provide mechanisms to support the integration of different levels of detail into a consistent holistic representation of a complex object. The availability of mechanisms supporting a consistent "analysis" at different levels of a DT will be very useful and will promote a better understanding and a larger usage of the DT technology.

A related but underestimated topic for service architectures and modeling of systems is the one of expressive power [113] (RT.19), i.e., the capability of software and its interfaces (APIs) to fully represent the capability of a (physical) system, its attributes and functionalities at different levels of details (but keeping the whole consistency). Having a methodology to assess and evaluate how much a software expresses the capabilities of a physical system would be a great support to the development of modeling and implementation of Digital Twin solutions.

These are some research topics that needs further investigation and a clearer definition of the problem domain.

A.4 Research over the Years

An attempt to provide a view about the overall trajectory of research over a long period is given in this section. Different organizations have different requirements that can affect how research is carried out and published. Discrepancies are the norm when a person moves from an industrial environment into an academic one.

A.4.1 Preamble

In spite the author's involvement in research for over 30 years, the activities conducted within the industrial sector have been kept internal to the company with very few controlled opportunities to be open to a wider community. The participation to conferences and congresses as well the publication to journals were limited and priority was given to internal reports and sometimes to enterprise technical journals and publications such as CSELT Technical Reports and Telecom Italia's Notiziario Tecnico. More recently it was possible to open up the research results and to contribute to discussions in a wider audience.

A.4.2 Research Fields

The research has touched several fields and topics. Ranging from initial attempts to use Artificial Intelligence for Natural Language Translation driven by semantics back in 1987 (Master thesis) up to application of Artificial Intelligence (a totally new AI!) applied to modeling of Digital Twin. In the following sections a light summary of these activities is given according to major research areas.

Service Architecture

This topic has been a major research field from 1987 on. It encompasses studies related to Intelligent Network [120], definition and implementation of devices for integrated com-

munication [121], [122] or remote update of Telecom devices. In the early 90ies of the last century (!), the focus of Telecom Operators was on the creation of Service Platforms for the provision of advanced features. A lot of interest was attracted by proposals like Telecommunication Information Networking Architecture (TINA) and the related activities of an international consortium comprising Operators and Vendors (TINA-C). The author was in the core-team of TINA-C and had also the responsibility of the Service Architecture group. Here major contributions were in the definition of Service Architecture baseline documents [123], [124], [125], and the TINA Book [126]. The paper [127] was a controversial contribution within the TINA community and caused some discussions about the directions taken by the Consortium about programmability and new models for communications.

The idea of introducing middleware for supporting the service architecture was maturing, but there were still proponents of older approaches based on Intelligent Networks [128]. In addition, the Internet was massively entering into play and this was a game changer [129], [128]. The trend towards middleware was emerging [130] and the TINA proposition had to confront with this and overall with the Internet. Between end of the nineties and beginning of 2000, the interests and investigation about Middleware platforms [131], [132] [133], [134], [135] was a clear research trend with particular interest to Application Programming Interfaces offered by platforms as Parlay (proposed by BT Group as an evolution of TINA)[136]. The concept of Next Generation Networks was also comprising programmable platforms for supporting new multimedia services [137]. In that period several Telecom oriented initiatives were undergoing aiming at bringing the Internet programmability to this industry. Also the Parlay initiative did not succeed [138] because of the emerging Web communications technologies. They were progressing so quickly that they took by storm the Telecommunication industry with particular impact on the services architectures [139].

The Next Generation Network Paradigm was involving many efforts of Telecom Operators for reshaping their control infrastructure. In that period, each major Service Communications Provider had a relevant project for leading to this platform transformation. Telecom Italia had its own called NOEMA ("intelligence" in greek). These activities were kept secret because considered very competitive. Actually, a lot of discussions and relationships with vendors was taking place to create new programmable control platform. The tricky point was that working with a vendor for developing a product meant that after 6 months, the other competing Operators could have the same product at a market price.

In that period, IP Multimedia Subsystem, IMS, was considered a key enabler for the provision of control and services in mobile and fixed network. A great effort was put on the definition of signaling capabilities. A Telecom Provider led initiative called 3G.IP was formed to contribute to the choice of next protocol for providing control and services. The 3rd Generation IP choose the SIP protocol over H.323 [140] and proposed it as the basis for 3GPP. The work of NOEMA project and the development of enterprise SIP based solutions were a relevant internal contribution. The IMS was, unfortunately, interpreted once again in the "Telecom way" and led to the definition of an architecture supporting session based services unable to bring to Operators the same flexibility offered by internet. The web companies were taking advantage of end-to-end offered by a large scale Internet. Some criticism already emerged in [141] that, later on, led to a set of studies about how to build

services and the differences between deperimeterized Internet vs. confined Telecom services [142], [143]. Actually, the change of perspective was due to a personal visit in Google that resulted in a paper [144] in which the role of network broker was clearly assigned to Google instead than to Telecom Operators. This paper was inspirational for the studies related to the creation of services and all the relevant web technologies. It was also internally attacked in the company because it was questioning the capability of Operators to stay on the service market.

Several papers were analyzing the differences between building services in a Telecom environment compared to the web one with particular attention to overlay networks [145]. In fact, the years between 2008 and 2009 were fruitful for a transition to studies related to more distributed and overlaid networks bringing a set of results in the field of adaptive networks [146], [147], [148], [149]. Also this approach and the achieved internal results led to attacks of the opponents of a move towards web technologies. A comparison between the constraints of telecom environment and the internet one resulted in a proposition of a new peer to peer autonomic service architecture [150], [151], [152] in contrast with the Service Delivery Platform proposed at that time [153].

These activities resulted in several research trends:

- The vision of an autonomic cognitive service platform made out of largely distributed resources scattered between edge and cloud [154], [151], [155], [156], [157], [158], [159], [160], [161], [162], [163], [164], [165], [166], [167], [168].
- The progressive virtualization of resources also in the telecom environment [145], [113], [169], [170], [171], [172].
- The need to adopt IT technologies for developing services and platforms [152], [173], [174], [175], [176], [177].

These studies have had a meaningful impact on international initiatives: the direct participation to the seminal work of ETSI about Network Function Virtualization [178]; the interest for "Clouds of virtual machines in edge networks" [179]; the participation to Soft-Fire Project⁴ for instance [172]. Surprisingly enough, Telecom Italia, at that time claiming the need for network transformations, decided to leave the only project it had about network transformation. And that was the end of author's work about the Service Architecture FOR Telecom Operators (but, ... that's another story).

More recently, the allocation of microservices in the edge cloud continuum was studied with respect to the latency issue. A simple mechanism for matching the applications requirements in terms of latency, and/or deployments costs was proposed [76].

Service Opportunities and Business Approaches

The need to study impacts and consequences of internet services over the telecom industry has generated a number of papers and contributions. First a new approach, based on the full adoption of ICT technologies and open to external contributions, has been proposed

⁴Some documents and results available in <https://softfire.eu>

[180]. It triggered discussions about a redefinition of the research model within Telecom Italia, but a more conservative approach based on testing was adopted.

Then the different span of Internet services and their supporting technologies over traditional Telecom ones has been considered as well as a number of paradoxes that this approach holds [181]. A deep analysis of differences between the technological and regulatory approaches of Web Companies and Telcos has been carried out emphasizing how the progress of software technologies was essentially due to the needs and efforts of Web Companies, while Operators were lagging behind [182], [183], [184], [185], [186], [187]. The key elements of this analysis were:

- The perimetrization of Telco Services within a proprietary network.
- The Telco's lack of software capabilities on a large scale.
- The centralized and "five nine"⁵ approach compared to Web companies that were distributing functionalities and were adopting different approaches to reliability.
- The business proposition based on freemium towards the "pay per service" of the Telcos [144].
- The need for Operators to choose a clear role (from service provider to bitpipe providers).

These topics and their technological aspects were fully analyzed in the Ph.D. thesis [188]. It has been deeply discussed within Orange and Telecom Italia contributing to some transformative attempts within these organizations. This document found its way towards the publication as a book [189]. The author was also one of the proponents of the adoption of virtualization and usage of SDN technologies within the company. The proposal had the support of the Strategy Director and the CEO of TIM at that time, but company high management did not embrace the same road. They created a large software center aggregating all the software competences of the research center into a large group without a clear and pragmatic development objective. This led to the dissolution of TIM's research center.

Especially the comparison with Google services was pushed ahead, but it was not understood immediately. For instance, the "end-to-end principle of the internet"[190], [191] and the consequent approach to peering from Google and other web companies was not comprised and it yielded to the loss of value on networks interconnections. The approach to fiber of Google⁶ was also brought up but to no avail.

All these changes in the Telecom industry have been emphasized internally and with some papers, but with scarce results for the Operator and some frustration for the Author.

⁵The ability of Telecom Network to operate 99.999 % of the time.

⁶Homes with tails, in which the ownership of last mile was given back to users, see <https://publicpolicy.googleblog.com/2008/11/homes-with-tails.html>.

Identity Management

A byproduct of the business models analysis was the proposal of a different approach to the personal data management. In this sector the proposition was to empower the user to have control of its identity (i.e., "names and identifiers") [44] and subsequently to propose a different platform and mechanisms for the management and protection of personal data [43]. This approach was promoted within the Strategy Department of Telecom Italia and resulted as one input for the definition (or redefinition) of GDPR policies. There was also a short discussion with a Browser Developer company for adopting a more protective mechanism for personal data safeguard, but it was rejected because it was too much a game-changer.

In spite of the generated interest, these activities did not led to a different approach within the Telcos.

Internet of Things and Digital Twins

The work and studies on service architecture found a logical continuation on the Internet of Things. The merge of communication means together with the possibility to convey data useful to provide services was instrumental to the definition of a powerful architecture putting together distributed processing, storage, communications and "sensing and actuation". On another research path, the possibility to virtualize things and the studies on Smart Cities conducted within Telecom Italia [192] were leading to the possibility of using the Digital Twin concept also outside the manufacturing domain. This research resulted in a number of papers and activities such as [193], [37], [194] that were instrumental to contribute to the IEEE Internet of Things Initiative. In this context a number of articles and white papers were produced [195], [196] and the widely cited [197]. The starting and consolidation of the World Forum on Internet of Things conference was a major result of these efforts [198] together with a number of special issues about Internet of Things on the IEEE Communications Magazine [199], [200], [201], [202].

The subsequent work on Digital Twin and its relation to IoT brought to the publication of a major article [30]. This is a particularly important paper because it started out the definition of properties that characterize a Digital Twin. A developers intended version of it was published as well [31]. At the same time, synthetic sensing studies were promoted by the publication of an article [80] and the special issue about Digital Twins in the IEEE IT Professional magazine[10].

The next effort in this field was to editing a Digital Twin book [12] comprising technological contributions and use cases from renowned experts of the sector (e.g., Michael Grieves [8]). Two contributions were addressing the possibility of extending the usage of DT to many application domains [12] and what the impact of the DT development and maintenance within enterprises and organizations is [20] as well as the relationship of DT and Artificial Intelligence [39]. Other contributions in the topics related to Digital Twin and IoT are [63], [32], [67], [65].

The work about Digital Twin modeling and development is going on with controversial submissions[49], [77] and others.

On the failure side, it is worth to mention the large effort, in which the author was deeply involved, about the creation of a Joint Research Center for Digital Humanities and Future Cities in Venice between Ecole Polytechnique Federale de Lausanne, Ca' Foscari

Venice University and Telecom Italia Future Center ⁷. The topics of Virtual Continuum (i.e., the Digital Twin) applied to Cultural Heritage, the city as a complex adaptive systems, City Monitoring and Ambient Intelligence were all anticipating some of the research paths to come. Unfortunately, a company restructuring and the leave of the CEO and the Director of Strategy Department left the agreement on hold.

On a similar path, several IoT centered activities for the city of Venice in a potential cooperation with the Italian CNR's Institute of Marine Research were considered but not consolidated. They were related to studies about flows in the Venice Lagoon and their impact on the City. The IoT and communication competences were instrumental to the capture of relevant data.

Smart Computing

Under the term of Smart Computing, a set of research activities are aggregated. The common point of them is the usage of Data Management or some forms of Artificial Intelligence techniques. It is a common trend of many IT related studies. In particular, there is a major author's interest on the application of AI to IoT, telecom infrastructure, and the urban environment. However other fields may be touched as well.

The cross fertilization between AI and IoT is a crucial point of study, especially with respect to understanding large phenomena in the smart cities or large environments. The sensed data are to be curated and properly used. They can also be related and operated in conjunction with a model of a system, i.e., a Digital Twin. Under this perspective a number of papers have been produced tackling different issues: smart city [203], [204], [28], [29], [55]; digital twin [39], [67] [72]; (optical) networks [205], [206], [207], [207]; data management [15], [5]; fitness, blockchain, anomaly detection and others [208], [209], [210], [211], [212], [213].

A.4.3 Oh, That Book

In 2020, Noel Crespi, Adam Drobot and the Author had the idea of editing THE DIGITAL TWIN BOOK[11]. The original definition of the book comprised less than 15 chapters, the final edition has almost 40!! It took a substantial effort and time in coordinating the different contributions and to normalize them. The story telling was another nightmare, the topics and contributions were spreading in all directions. The harmonization was another daunting task. However, at the end, the result was worthwhile and rewarding. The Editors promised not to do it again, but another book about DT Platforms is on its way (March - June 2025).

A.4.4 Open Contributions

A number of open contributions have been provided along the years. A few of them are worth to be mentioned:

- The set of Architecture and Components specifications related to TINA Service Architecture [123], [124].
- The thesis [188] about evolution of Telcos⁸.

⁷<https://www.gruppotim.it/en/press-archive/corporate/2013/02-23-is.html>

⁸available at: https://theses.hal.science/tel-00917966/file/MINERVA_Roberto-2.pdf

Annex: Research Background

- The set of documentations [197] and instruction material⁹ created within the IEEE IoT Initiative.
- Various projects' deliverables such as [68] or the SoftFire document set¹⁰.
- The important contribution about Network Function Virtualization [178] that was instrumental to ignite the ETSI's initiative on Network function virtualization.

These documents have had an impact on the Telco community with some reflections on the academy.

A.5 Additional Information

In this section some information about teaching and metrics of publishing is provided.

A.5.1 Teaching

The author is involved in several teaching engagements. In particular in the following courses:

- CSC 7016 Internet of Things and Digital Twins: 18 h.
- CSC 7223 Distributed Computing Systems: 21 h.
- NET 4202 Smart Computing: 20 h.
- NET 4252 Multimedia Services and Networks: 20 h total.
- NET 8562 IOT et Big Data: coordination and 10 h.

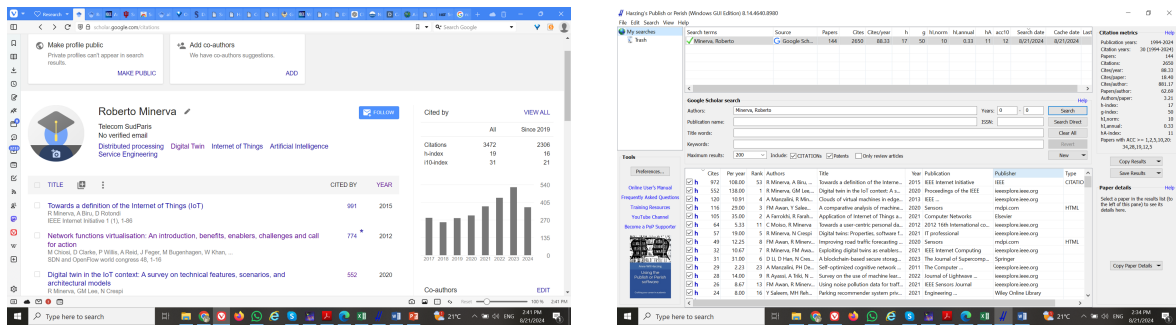
These courses are updates annually to present students with actual results and new research challenges as well as new industrial applications. In the past, the author contributed to some cycles of lectures for the Politecnico of Turin and courses held at the Scuola Suoeriore Guglielmo Reiss Romoli in L'Aquila, Italy. It was the formation school of the Telecom Italia group.

A.5.2 A Bit of Metrics

The measurement of impact of research is a controversial topic. In this particular case it is worth to recall that the author's academic path has not been a linear one and it has been influenced by the work done in a large enterprise with strict publishing constraints. In spite of this, A.13a and A.13b show an increasing H-Index especially in the latest years. This is not surprising because there are a few peaks in scientific production: one during the TINA-C period (around 1995); another one in conjunction with the possibility to express concerns about the Service Delivery Platforms (around 2000); one related to studies and publications for the Ph.D. completion (around 2013); and a smooth increase from 2018 on, when publication was freer and an objective of the organization (i.e., Telecom SudParis). Two publications, i.e., [197] and [178], count together 1765 citations over a total of 3473 (on August 21st, 2024). However, there are some important numbers also for [30] with over

⁹<https://iln.ieee.org/public/contentdetails.aspx>

¹⁰A Federated Testbed for Future Networks (and 5G), 2018, Available: <https://www.softfire.eu/>



(a) Google H-Index Calculation.

(b) Harzing's Publish or Perish Analysis.

Figure A.13: Analysis of Research Impact

500 citations and [179], [28], and [208] with more than 100 each. The article [43] is still attracting interest and it has been cited by 64 articles, with an average citation number of 5.33 per year, after a long period of time. Actually it was an anticipatory paper addressing the issue of personal data ownership.

The number of papers is increasing over the years, but the chosen approach is to enable (Doctorate and Master) students to tackle important research topics with an impact on industry. The goal is to exploit the experience gained over the years to support new researchers in finding their path into academy and industry by tackling relevant topics.

As a sort of complementary tool to H-Index, Scholarometer has been taken into consideration: it is "a social platform to facilitate citation analysis. By providing author field annotations, the tool uses crowdsourced data to evaluate the impact of that author's publications. The impact metric can be compared across fields". Scholarometer can be installed as a browser extension. Fig.A.14 shows results and a relation map. The next wave of re-

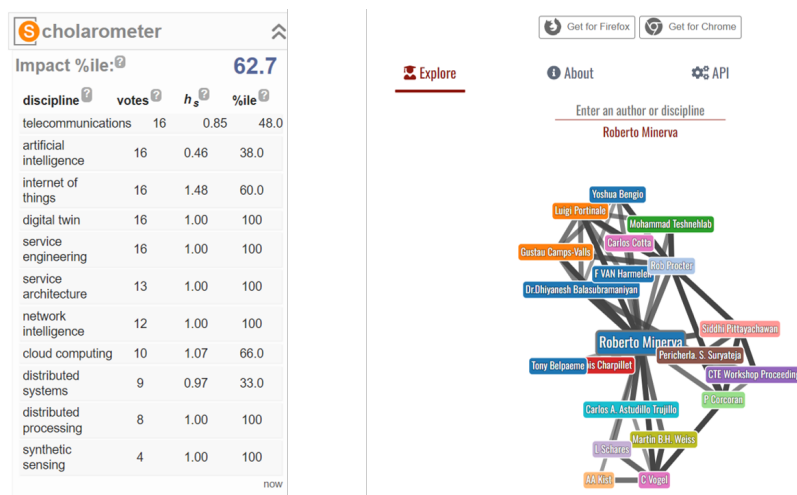


Figure A.14: Scholarometer Results and Relations Maps.

sults is expected by the co-supervision of Students involved in Digital Twin, Smart City, and General Purpose Sensing studies.

A.5.3 Mentoring of Students

With this approach, a number of Students have been supported in their path towards the Doctorate or a Master Degree. From 2018, three co-supervised students have obtained their Ph.D. They are:

- Yasir Saleem Shaikh, now with Aberystwyth University, UK;
- Faraz Malik Awan, now with Urban Big Data Centre in Glasgow, UK;
- Reda Ayassi, now Research engineer - Huawei Technologies, France.

Other are currently pursuing the Ph.D. goal under co-supervision. They will achieve their objectives in the next couple of years.

- Syed Mohsan Raza, working on Network Digital Twin and microservices architectures.
- Herath Hemawanshage Chethinga Manoj, working on DT architectures and DT Modeling.
- Maira Alvi, working on Smart Cities and Digital Twin.
- Aung Kaung MYAT, working on general purpose sensing and bottom up DT.
- Ranjit Kannappan on a joint doctorate with Orange Innovation.
- Leyuan DING, working on Redundancy of sensors and smart computing.

The author's HDR path will be instrumental to even better support current students or new ones. The roadmap presented in 6.5 is instrumental to frame new students and their activities into a consistent flow of research that will exploit existing results (of previous students) and will drive towards new achievements. Obviously students' inventiveness will play a major role and will pave the way to new paths and results.

A.6 Acknowledgments

This research trip has taken more than thirty years from the initial steps in CSELT and then Technopolis to the current vibrant moments within Telecom SudParis. I've met many great researchers, but what is most important, many times these great researchers were and are beautiful people.

I want to thank each of them starting by remembering Roberto Saracco that, in the darkest period for research in Telecom Italia, supported and introduced me to the richness of IEEE. Thanks Roberto for this and much more you have done for me.

Two road companions should be mentioned from the CSELT/TILab period: Antonio Manzalini and Corrado Moiso that triggered a lot of discussions and new ideas that, sometimes, found their way into the network and, sometimes, into good papers, and most of the times in heated technical discussions.

Domenico Rotondi, from Technopolis, taught me the rudiments of research. I hope I learned well enough not to disappoint him!!

I need to thank Susanne Kuehrer and Fabio Pianesi of EIT Digital for the support and help during the Softfire project. They were instrumental to a life change.

And what to say about Noel Crespi and Djamel Zeghlache? There are not enough words to say thank you for giving me the possibility to do research (at my old age !!!). Thanks to Natalia Kushik and Hang Nguyen for their precious suggestions and support!

And thanks a lot to Praboda Rajapaksha (I saw you becoming a well respected professor starting as a shy student), Reza Farahbakhsh (that has been so kind in supporting me all over these years) and Yasir Saleem (helping me with projects and many many activities); and Emmanuel Bertin from Orange for a "distance" support always present.

And then other doctoral students: Koosha Zarei, Faraz Awan and, later on, Amir Reza Jafari that pushed for AI technologies and increased my understanding. I secretly called them "the old guards" :). Thanks also to Reda Ayassi for his dedication and willingness to complete his work. Thanks for the great moments and results shared.

A special thank goes to the current DICE team: Syed Mohsan Raza, Maira Alvi, Manoj Herath, Aung Kaung Myat, Razieh Chalehchaleh, Leyuan Ding, Shreya Sarker, Dun LI (Eric), Sergei Berezin, Khoulood Mnassri, Ranjit Kannappan, Guanlin Li, and post-doc Hrishikesh (Rishi) Dutta (you'll do great in such a team!) and all the others that I certainly forget to mention (have mercy for an old grumpy researcher :). I know I learned a lot from each of you and I owe you a lot. I hope to have instilled in you a bit of passion for research and the need for perseverance. Your paths will be brilliant.

I have a great debt towards Professors and Students of University of Calabria, namely Prof. Giancarlo Fortino and Prof. Claudio Savaglio, and the "Italian Pack" (as I called them): Vincenzo Barbuto, Fabrizio Mangione ("Daje Roma") and the Stefano's, i.e., Stefano Morrone and Stefano Perna. Without them, the GPS experiments would have been much less.

I also express my gratitude to the Reviewers of this work: Prof. Luigi Atzori, Prof. Roch Glitho, and Prof. Stefano Secci that helped me with comments and suggestions to improve the thesis. And to the Jury members that kindly agreed to evaluate this work.

Last words for my kids, it was tough to see you in trouble and in the hospital for so long. I wanted to give up this work so many times, but you also gave me the strength to continue. Thanks with Love.