Università degli Studi dell'Insubria
Dipartimento di Scienze Teoriche e Applicate

Privacy for Human Digital Twins: Enforcement of Privacy Preferences in Complex Scenarios

By
Giorgia Sirigu

Supervised by
Prof. Barbara Carminati
Prof. Elena Ferrari

Thesis submitted for the degree of Doctor of Philosophy in Computer Science and the Mathematics of Computation - XXXVII Cycle - in the Department of Theoretical and Applied Sciences at the University of Insubria

Varese, Italy
February 2025

Declaration of Authorship

I, Giorgia Sirigu, hereby declare that this thesis and the contents presented in it are the results of my own original research, with the exception of where specific reference is made to the work of others. I confirm that any part of this thesis has been submitted for a degree or qualification at this University or any other institution. Therefore, I declare that this thesis was done wholly on candidature for a PhD degree at this University and that it is based on work done by myself, with the support of my supervisors, for whom I am grateful.

Dedication

To my Parents, who have supported and tolerated me throughout my academic journey as a student. It has been a long road, and you have always been there for me. Thank you, I hope you are proud of me.

To All who have taught me something in any field - academic, life, and more: a degree is not the only essential element. Your knowledge and experiences have modelled who I am today and have been essential in achieving this important goal.

To All people who cannot follow their dreams for reasons beyond their control. To those who are unable to achieve their objectives. To those who are forced to follow a path and cannot choose their own. I hope all these people can take control of their lives one day.

To Myself, the person I was when I first started this journey as a bachelor's student, wondering what it would be like to earn a PhD. Now, I can proudly say *I did it.*

Acknowledge

There are many people who I would like to thank for this significant accomplishment. While some have been with me from the beginning of my PhD journey, others have been part of it for a shorter period. Regardless of when they entered or left, their presence and role have been meaningful in shaping my path.

I am deeply grateful to Professors Barbara Carminati and Elena Ferrari, who made this achievement possible. Any part of it would not be possible without their expert suggestions and guidance. They have been an inspiration to me, and this has led me to do my best. I would also like to express my special thanks to Professors in Computer Science at the University of Insubria. They imparted to me their knowledge and fascinated me with their subjects during my Bachelor's and Master's Degrees. Without their contributions, I could not have acquired the knowledge that led me to write this thesis. Many thanks to Professors Bikash Chandra Singh and Lili Nemec Zlatolas for their thoughtful review of this thesis and provided me with valuable suggestions for improvement. A big thank you to Roberta Viola, Mauro Santabarbara, and the University's administrative offices for their constant support with technical and administrative tasks.

During my PhD, I had the privilege to meet new friends, researchers, and colleagues, who have contributed by adding wonderful facets to this experience. First, I want to thank Dr. Ahmed Lekssays, who convinced me to start my PhD. He played an essential role with his warm, kind and always supportive company. His way of being, behaviour, and calm yet determined approach to challenges are admirable. Thanks to Dr. Xuan Ha Son and Dr. Anh-Tu Hoang, I cannot forget their kindness in helping me at the early stages of my journey. I thank my lab mates, Hidawy Khaoula and Nguyen Tran Thanh Lam, for sharing our thoughts, challenges, and mutual understanding. I am grateful to Dr. Jesus Fernando Cevallos Moreno for teaching me to always find the positive in every situation. I also thank Mirco Gallazzi for sharing his experiences with me and always being there to listen. Last but not least, I would like to express my gratitude to all my fellow PhD colleagues at the University of Insubria. They have taught me that being a PhD does not mean isolating yourself in your research. Instead, being a PhD means sharing ideas and moments of lightness and growing together as a group.

A heartfelt thanks go to my family, whose support was a cornerstone throughout this journey. Although they could not completely understand what I was

doing, their curiosity, interest, and pride in me have been constant sources of motivation. These were pivotal in the pursuit of my PhD. Many thanks to my Mum for her patient care over these years, which has been essential to this achievement. I am thankful to my Dad for always being by my side when I needed help and for his unspoken pride in me. I am also grateful to my brother who, with his enthusiastic curiosity in my research, has let me feel that I was doing something that truly mattered. Dulcis in fundo, a huge thank to Jack-Ino. His irreplaceable presence and his intrusiveness (with affection) in my life have been a godsend.

This last part, yet not the least important, is dedicated to my friends, adventure companions, and all with whom I share my passions. I want to thank them all for the time spent together, the laughter, and their support and encouragement in the most challenging moments. Their support was essential and boosted me to do my best to continue and terminate this journey as a PhD student.

"Data privacy means empowering your users to make their own decisions about who can process their data and for what purpose."
- European General Data Protection Regulation

Privacy for Human Digital Twins: Enforcement of Privacy Preferences in Complex Scenarios

Giorgia Sirigu

Abstract

The advent of new technologies has brought new features, improving opportunities and making services more attractive. Human Digital Twins (HDTs) are a novel technology generating a digital representation of an individual [1]. HDTs are implemented in different areas, such as healthcare, to monitor disease and test therapies, in Industry 4.0, to improve the workers' well-being, or in the Metaverse to generate an accurate avatar. Since they require extensive personal data for these tasks, they have become an attractive target for hackers. Therefore, protecting users' privacy and limiting data sharing is fundamental for users who want to use HDTs. This thesis aims to ensure privacy policy enforcement when employing this technology. To this aim, we first investigate HDTs in terms of security and privacy. Then, we present ConPrEF, a framework for edge computing - a paradigm that aligns with HDTs (e.g., edge nodes acquire users' sensed data for generating the HDT) - where users can select their privacy preferences based on the contexts, namely, time, location, activity, situation and social interaction. Another challenge is the time required for HDT materialisation when privacy preferences need to be enforced, leading to delays in HDT-based services. This need improves when HDTs are used in complex services, such as processes, where each task involves a different service provider. Therefore, we introduce a system, HDT-ViewMat, which identifies the parts of an HDT that should be pre-materialised for each task, depending on the user's chance of executing it. We then revise this strategy to consider more complex HDT-based scenarios, where multiple users jointly execute the process, and multiple providers deliver each task. Here, deciding which HDT portion needs to be pre-materialised must consider multiple privacy policies and preferences. Finally, we surveyed a group of users to assess their agreement with the proposed solution, and we tested our systems to assess their feasibility and the benefits they bring.

Contents

List of Figures

List of Algorithms

List of Tables

Introduction

The demands to integrate digital technologies across society and the economy are growing, supported by the people's needs to improve the quality of their lives and access to services. The digital transition that has been characterising the last years has introduced new features and opportunities, such as smart working, ease of communication with friends and family members, and the convenience of e-commerce platforms, to name a few. The benefits of this change can be summarised into a better quality of life, empowering people to organise their days and carve out more time for themselves, their activities and social life.

On the basis of this transition, there has been technological evolution and digitalisation. A novelty is the introduction of Digital Twins (DTs): virtual copies of physical entities, industrial processes, or systems that can be employed for various tasks, including controlling, monitoring and analysing the status of the real (aka physical) twins. Thanks to these features, DT technology finds application in different areas, especially manufacturing and smart grids, and its market size is expected to reach USD 259.32 billion by 2032 [2].

The evolution of DTs has extended their scope to humans, leading to the development of the Human Digital Twin (HDT) [1]. HDTs mirror humans' psychophysical aspects, providing them with the same benefits DTs introduce for physical objects [3]. The main difference between modelling DTs and HDTs is that the latter is limited not only to the physical aspect of a person but also to human beings (e.g., the psychological state and the environment). The employments for HDTs are various, but the most common are in healthcare, where it is used to monitor disease and the effect of a therapy or to test drug administrations before giving them to the patient. Another example of use is in Industry 4.0, where workers are tracked to improve their well-being, safety and productivity. Still, HDTs are used in the Metaverse to model an avatar that is as representative as possible to the user.

(H)DTs require constant synchronisation with the physical counterpart in order

to have a model that reaches high degrees of accuracy - namely, the virtual copy should be as close as possible to the real one to make precise predictions and correct decisions. This achievement requires a huge amount of data to reduce divergences between the physical and virtual words. Data concerning HDTs are, for instance, sensed data from wearable devices and medical reports. It follows that copious sensitive data are involved in HDTs, and thus, they become extremely attractive to hackers, who might attempt to steal data to ask for a ransom or sell it to a third entity. This illicit acquisition and sharing can cause serious damage to the data owner who would see all his/her data disclosed without any authorisation. As a further consequence, a victim could be discriminated, for instance, against at work (e.g., he/she cannot get a job or is unfavoured during political elections) or in social relations (e.g., friends would interact with the victim differently once they discover he/she has a determined illness). Therefore, privacy in HDTs is one of the main concerns due to the sensitivity of data required for making an accurate model. In fact, guaranteeing privacy is essential to prevent any illicit disclosure, ensuring a user with secure data handling. Generally, sensitive data, like the one in HDTs, must be protected in accordance with current laws and standards (e.g., GDPR, HIPAA, and CCPA). Service providers must declare their privacy policies to inform users of how they will collect, manage, and disclose their personal data, and they should only have access to the minimum data necessary for their purposes. In the scope of HDTs, this constraint translates to service providers accessing a limited part of the HDT, called *view*. On the other hand, people have to specify privacy preferences to limit their data disclosure. In addition, different studies (e.g., [4] and [5]) demonstrate that privacy concerns depend on the context, meaning users can have different preferences depending on factors such as their location and the time. It is, therefore, important to allow users to specify their privacy preferences to adapt to the current context since their privacy needs are inherently dynamic, changing based on location, activity, social environment, and time. For instance, a person might intend to share personal health-related data (e.g., medical reports on the cardiovascular system) for a medical examination in a hospital and not in any other place (e.g., in the office).

Another issue concerning HDTs is the time required for their materialisation [6]. This process consists of making computations on personal data to generate a digital representation for service use. For instance, a patient with heart disease needs his/her cardiovascular history in a human-readable form to let the cardiologist perform the examination. Depending on the accuracy and quantity of the data employed, this operation requires time as it may involve heterogeneous data

acquisition, parsing, and processing to make the data functional for the data consumer and, if needed, creating a 3D view (e.g., the generation of a Cardiac Digital Twin might take four hours [7]). Generally, data materialisation leads to delays in HDT-based services. This delay would create discomfort for the users who want to use their HDTs for a service, especially if time is essential. For instance, if an individual has a severe incident, the HDT could be essential to save his/her life. However, waiting hours for its materialisation would make all the positive effects derived from the use of this technology disappear. This issue will worsen in more complex services, consisting of workflows/processes of multiple tasks, each requiring a different HDT view materialisation, and/or when privacy preferences of end users are taken into account.

Thesis Contributions

The contributions of this thesis, which are depicted in Fig. 1, are related to the definition of privacy enforcement mechanisms for HDTs.

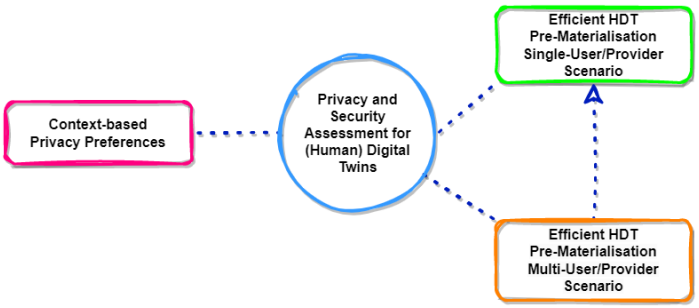


Figure 1: An overview of the contributions of this thesis.

The central element is the analysis of security and privacy issues in DTs and HDTs that we made in a visionary paper [8]. In this work, we identified the security and privacy issues affecting this technology and discussed state-of-the-art solutions. Then, we focused on the open research challenges in the field of HDT's security and privacy.

An additional main contribution of the thesis is about the definition of privacy preferences and their enforcement, considering the context where users' data is collected. This is a critical aspect in dealing with privacy applied to HDTs due to the sensitivity of the involved information.

We first focus on enforcing context-based privacy preferences in edge computing environments [9]. This paradigm is characterized by edge nodes, which are machines (e.g., routers) located at the edge of a network, close to the users. These nodes are provided with computation capability and act as intermediaries between a user's device and a remotely hosted service provider, enabling faster data processing. The decision to first focus on this paradigm is driven by the fact that it has been studied more in the literature than HDTs. In addition, edge computing lends itself well to the implementation of context-based measures because it is a dynamic environment (e.g., users move and change their location). Furthermore, HDTs can be reconnected to edge computing because individuals' data is acquired by sensing devices that interact with edge nodes acting as intermediaries for computations or storage. Hence, guaranteeing data privacy in this scenario, such that edge nodes collect only the data the users agree to share, is a key element for protecting privacy for HDTs. The proposed solution, called ConPrEF [10], is a framework that allows users to specify their privacy preferences in different contexts. For instance, an individual may want to share medical health records with the edge node only at the hospital or in an emergency. We model context by including location, time, type of activity, situation, and social interaction (i.e., limiting data sharing in the presence of some people, like a family member). All fields are almost effortless to check, but the latter is the principal challenge for ConPrEF, as it requires the user's device to identify who is connected to the edge while maintaining the privacy of the user himself/herself and of who is connected. To cope with this requirement, we adopt a private operation mechanism which requires time for its computation.

The other main contribution of the thesis concerns the pre-materialisation of the HDT views, whose principal purposes are twofold: limiting personal data sharing considering the transmission of views instead of the whole HDT to service providers, and regulating the HDT materialisation to avoid waste of time and resources. In fact, a naive solution to this issue is to pre-materialise the whole HDT in advance, that is, before launching the HDT-based service. However, this would compute views that may not be used, leading to a waste of resources. Therefore, there is a need for strategies for identifying the views that should be pre-materialised, taking into consideration the trade-off between potential delays and resource waste. To meet these requirements, we developed a system, HDT-ViewMat [11], that enforces efficient HDT pre-materialisations in complex services. More precisely, we target a scenario consisting of a user who wants to use their HDT in a workflow where each task is provided by a service provider that states

its privacy policies, defining the data it requires to supply the service - aka the data that will compose the required HDT view. In order to ensure timely pre-materialisation and avoid consequent waste of resources, HDT-ViewMat selects which views for the tasks in the workflow must be pre-materialised before the user requires them. This strategy involves analysing the process requiring HDT data and considering the compliance of privacy policies with users' preferences to avoid pre-materialising views that users do not want to release. This scenario is further complicated if we consider that a complex process typically involves multiple providers - i.e., one or more providers for each different task - with different privacy policies - and multiple users - with different privacy preferences. In this circumstance, having a single policy for each required data is more acceptable for users, who do not have to select their preferences multiple times. Hence, we provide a negotiation mechanism that identifies a representative policy that considers the providers' policies and the users' privacy preferences. For instance, if users are required to share their physical aspect with a number n of service providers, they would need to accept n policies. Instead, a representative policy would enclose all providers' requirements, allowing users to give their agreement only once. In addition, the representative policy should also consider the users' privacy preferences so as not to be too demanding concerning their preferences. Finally, we did a survey to analyse the participants' satisfaction with the proposed representative policy, aiming to understand whether the negotiation mechanism could increase their propensity to share their data under a policy closer to their preferences.

Thesis Chapters

The thesis is organised as follows:

Chapter 1. This chapter introduces the main concepts on which the thesis relies, namely privacy and Human Digital Twins. Concerning the former, it explains the challenge behind finding a universal definition of privacy and then reports a historical overview of privacy regulations, focusing on the European Union. Next, we introduce HDT technology, architecture, and application areas.

Chapter 2. The security and privacy challenges affecting DTs and HDTs are presented in this chapter. Then, it focuses on HDTs and on the challenges raised by this specific technology.

Chapter 3. In this chapter, we analyse related literature about security and

privacy practices to protect (H)DTs. We also include privacy policy negotiation mechanisms and proposals to protect users' privacy in edge computing.

Chapter 4. In this chapter, we describe ConPrEF, a framework to protect users' personal data before sharing it in the edge computing environment, which considers contextually defined privacy preferences.

Chapter 5. This chapter illustrates HDT-ViewMat, the framework enforcing privacy preferences for users using their HDTs in complex services. It also includes an application of HDT-ViewMat in a more complex scenario characterised by multiple users and service providers, which requires the negotiation procedure. At the end, it illustrates the survey we have designed to assess the users' appreciation of our framework.

Chapter 6. Finally, we conclude this thesis by summarising the addressed requirements and our achievements. We also discuss future work to improve the designed frameworks.

1 Background

This chapter provides a description of the two principal concepts over which this thesis relies, namely privacy and Human Digital Twins. Although the former is fundamental in the right to protect personal data, it lacks a universal definition. This absence is due to different perceptions that various cultures and societies have regarding privacy. In addition, privacy aspires to protect personal data (aka personal information), but determining what information this concept includes is challenging. For instance, must only data strictly related to the individual be considered as personal data (e.g., name, race, and gender)? What about information related to the ideas (e.g., politics and religion) or physical and psychological status (e.g., heartbeats and blood pressure)? Moreover, with the advent and improvement of technology, further data needs to be included in the sphere of personal data (e.g., IP address).

For what concerns HDTs, this technology is a novel evolution from Digital Twins (DTs), which represent a virtual copy of an entity (a human in the case of HDT). These technologies have many common elements, such as the general architecture composed of a physical (sensing) layer, a virtual layer and a communication interface. While DTs are mainly used to model organisations' assets (e.g., robots), HDTs represent humans or part of them (e.g., systems). Both require constant synchronisation and extensive data (e.g., psychophysical data for HDTs)

to generate an accurate representation.

Privacy enforcement is essential to protect HDTs, as well as their owners, due to the typology and the quantity of data required for accurate human representation. An HDT may become a sensitive target for any adversary capable of illicitly accessing it. Therefore, when users use their HDTs, they must have strong privacy guarantees.

In this chapter, we first introduce the meaning of privacy and how it has evolved over time (Section 1.1). Then, we focus on Human Digital Twins (Section 1.2), analysing, at first, DTs, then HDTs and their application scenarios.

1.1 Privacy

Privacy is a multifaced concept that encompasses several key aspects belonging to the private sphere of an individual, aiming to keep personal data private. It is a complex paradigm that varies over time following technological evolution - which improves the efforts needed to protect it - and across different cultures - which have different requirements depending on the individual and society values [12]. Due to this continuous transformation, providing a formal definition takes time and effort, as well as defining what personal data is to be protected.

As [13] states, the first definitions of privacy were related to the single person and were about data disclosure or confidentiality. While the former concerned the agreement to share personal data, the latter consisted of maintaining control over access. It is also important to point out that the decision to share data depends on the gain for the data owner. Namely, this decision is a trade-off between personal benefits and data loss.

Technological evolution brings new information types and ways to collect, store, and share them, enlarging the sphere of individuals whose privacy must be protected. For instance, let us consider the case of a social network user posting his/her photo with some friends. This circumstance is no longer a matter of the single individual (i.e., who posted the photo) but affects the privacy of a group of users (i.e., friends depicted in the photo). Moreover, context has a significant role in privacy decisions. In fact, a person may make different decisions depending on multiple factors (e.g., the conditions or the entity with whom data are shared) [14].

This section details these privacy aspects and a historical view of the concept of privacy.

1.1.1 Privacy Components

It is clear that privacy is universally demanded and that some key elements characterising its definition are individuals' social, economic, and cultural characteristics [15]. In this matter, researchers distinguished between individualist and collectivist cultures while striving to find a definition of privacy [16]. The former is based on the assumption that an individual is independent of the others. People in individualist cultures prioritise themselves, their goals, and their choices over those of the community, placing themselves at the centre (e.g., European and American citizens) [17]. On the other hand, collectivism assumes that individuals are bound by social groups and roles. Collectivist societies centralise common values and goals, and the individual is a part of the group (e.g., Chinese culture) [17]. Based on this distinction, people belonging to these groups have different perceptions of privacy. For instance, collectivistic cultures are more concerned about damages caused by sharing others' personal information, contrary to individualistic cultures [18]. Also, Western countries consider medical data more sensitive than Eastern countries [19]. As a consequence, it is hard to establish a standard definition of privacy when the boundaries imposed are so different between different populations.

A further element affecting privacy concerns is the context where personal data sharing happens. For instance, some people decide whether to share their data depending on the service they get in return, or on the service provider. The authors of [20] conducted a survey on 111 students and discovered that they were more confident with the collection and use of their personal data in the university context rather than in an e-commerce context. Also, [13] surveys 500 Flemish about their concerns to share data collected from wearable devices. The results demonstrated that the principal contextual parameters that affect the willingness to share data are information type and purpose. For instance, people find sharing their weight and position less appropriate for research purposes, but not heart disease risk data. The authors justified this choice in the utility of discovering heart disease, which exceeds the need for privacy.

Overall, the concept of privacy pivots on the definition of 'personal data' - aka 'personal information'. However, it may be hard to identify data included in this category because it has evolved over the years simultaneously with technology [21]. For instance, while in the first definitions of privacy - namely in the twentieth century - personal data were mainly confined to the identity (e.g., name and surname), physical status (e.g., diseases) of an individual, and personal beliefs (e.g.,

religion and gender identification), after the advent of the internet, more data have to be included [12]. If we consider the issue of being monitored, in past years, it was a matter of being observed by other people. Meanwhile, today, this activity means that our actions are recorded and stored somewhere [12]. Notable considerations should be made on the spread of social networks, as they have introduced new personal data types, such as pictures and posts. Therefore, privacy in the era of digitalisation is more complicated because personal data may be collected and shared with third parties without the consent of the owner, users' profiles may be improved with sensor data, and there is a continuous information disclosure [22].

In general, data protection laws provide a broad definition of personal information, which consists of any data related to an identifiable individual. An example is the Canadian PIPEDA [23], which states:

"(...) personal information includes any factual or subjective information, recorded or not, about an identifiable individual."

Europe, through GDPR (see Section 1.1.2 for further details), extends this notion with the concept of 'identifiable information':

"Different pieces of information, which collected together can lead to the identification of a particular person, also constitute personal data"

Due to the generality maintained by the regulations, extracting which data belongs to this category is hard, and it will become more complex in the future [24]. Moreover, combining this ambiguity with pervasive global networks (e.g., IoT and cloud computing) and the constant growth of data generation, aggregation and analytics, all information could be defined as 'personal', making impracticable data protection [25]. Another aspect considered by privacy laws is the level of sensitivity of a data category. Sensitive data is a particular type of data that requires more protection because the damage caused by its disclosure is higher than that caused by less sensitive data. In other words, the risk associated with the leakage of sensitive data differentiates this category from the others [26]. For instance, according to the California Consumer Privacy Act (CCPA) [27]:

"Sensitive personal information is a specific subset of personal information that includes certain government identifiers (such as social security numbers); (...) genetic data; biometric information processed to identify a consumer;

information concerning a consumer's health, sex life, or sexual orientation; or information about racial or ethnic origin, religious or philosophical beliefs, or union membership. Consumers have the right to also limit a business's use and disclosure of their sensitive personal information."

To summarise, the concept of privacy includes social and ethical aspects that vary among the different cultures. In addition, it has to adapt to the technological evolution that results in an increasing variety of data types that must be protected, which have different degrees of protection depending on their sensibility. Considering these aspects, the difficulty of finding a universal definition for privacy rises, and all countries singularly specify internal regulations reflecting their needs.

1.1.2 The evolution of European Privacy Legislation

Despite several attempts, a universal definition of privacy has yet to be agreed. Warren and Brandeis were among the first to recognise this challenge when, in 1890, they described privacy as *"the right to be let alone"* in their famous seminar work *The Right to Privacy* [28]. Through their article, the two lawyers aimed to highlight the need to regulate unprecedented innovations that jeopardised people's privacy in that period, such as instantaneous photography. In particular, journalists of their era exploited these new means to fill newspapers with gossip by publishing pictures that broke the limits of the depicted people's private lives without their consent. With their statement, Warren and Brandeis complained about the right to protect their emotions and private thoughts as the right to protect their private property. Despite The Right to Privacy being the first article demanding the right to privacy as a right against emotional suffering [28], it is insufficient to address the complexity of modern privacy requirements. Indeed, the rapid evolution of technology (e.g., AI, storage media, and social networks), new concerns (e.g., data collection, retention and analysis), and global exchanges necessitate more comprehensive legal frameworks to protect privacy rights.

The European Union (EU) began posing attention to privacy concerns in the *European Convention on Human Rights* (*ECHR*) in 1950 [29], which states in Article 8:

"Everyone has the right to respect for his private and family life, his home and his correspondence."

However, this article did not include all technological and social facets related to privacy. Later, the Council of Europe adopted *Convention 108*, the first international data protection treaty [30]. It protected European citizens from abuses in collecting and processing their data; on the other hand, it restricted the transmission of this data to extra-European states with weaker regulations. Among the novelties introduced by Convention 108, there were the definitions of personal data and special categories (e.g., religion and political choices), the concepts of purposes and retention period, the right to track the collected data - i.e., which data, for which purpose and information about the collector - and the right to obtain the erasure if data have been processed contrary to local laws. This convention still influences the current European Privacy regulations, and the modernised version is called Convention 108+, updated with the latest requirements, principles and regulations.

After the entry into force of Convention 108, there has been a sequence of directives and regulations that legislate EU institutions, offices, and bodies, as well as regulate personal data procession and movements.[1] In 2000, data protection and privacy became fundamental rights in the EU. Subsequently, new privacy figures, such as the European Data Protection Supervisor (EDPS) in 2001 and Data Protection Officers in 2004, have been introduced. Later, the EDPS expanded to create the Supervision and Enforcement Unit, the Policy Unit, and the Consultation and Human Resources, Budget and Administration Unit in 2011, followed by the Information and Technology Policy Unit in 2012.

Despite the enormous progress made on the privacy rights of European citizens, data shared outside the EU's frontiers did not benefit from the same protection level. In 2000, the European Commission made the *Safe Harbor Decision* [31], according to which the US companies that adhered to the International Safe Harbor Privacy Principles - defined in agreement with the US Department of Commerce - could acquire personal data from EU citizens and transfer it to their local servers. Specifically, the principles stated in the Safe Harbor were seven: notice, choice, onward transfer, security, data integrity, access, and enforcement. Membership in the Safe Harbor Decision was voluntary, and the participating companies were obliged to enforce a satisfactory level of data safeguarding. One of the main considerations about this agreement over which the EU community was sceptical was the lack of a defined authority to ensure that the US companies were compliant with the requirements. Indeed, the participants had to self-certify their compliance, but there was no control over their actual enforcement of the privacy principles. The

[1] `https://20years.edps.europa.eu/en/history/timeline`

Safe Harbor Decision lasted fifteen years until two scandals hit US corporations in 2015, and the EU Court of Justice revoked the Safe Harbor Decision. The first was raised by Maximillian Schrems,[2] who discovered that Facebook had been illicitly transferring users' profile data from Irish to US servers since 2008. The other was a revelation by Edward Snowden in 2013,[3] who claimed that the US National Security Agency had illicitly been accessing the personal data of EU citizens under the guise of national defence. However, their activities exceeded what was strictly necessary to ensure national security. Hence, in 2015, the EU Court of Justice analysed how US authorities accessed personal data and defined their actions as "incompatible" with the purpose for which data was acquired and transferred [32].

After the decision to declare invalid the International Safe Harbor Privacy Principles, the European Commission and the US Department of Commerce had been negotiating to regulate personal data exchange for commercial purposes until they reached an agreement in 2016 with the stipulation of the *EU-US Privacy Shield* [33]. With this framework, the EU Commission intended to improve the safeguard of citizens' fundamental rights when their data is transferred to companies in the US, which had to ensure an adequate level of protection. Safe Harbor Decision and Privacy Shield shared almost the same principles; the main differences lie in the emphasis on personal rights for EU citizens and the requirements for US companies and government entities. After this agreement, the US organisation that wanted to get data from EU citizens had to self-certify to the Department of Commerce that they met high data protection standards annually. In turn, the US Department of Commerce was in charge of monitoring and surveilling the organisations that joined this covenant to ensure their compliance. However, the European Court of Justice invalidated the EU-US Privacy Shield in 2020 because it did not ensure the same level of personal data protection required by the EU law.

Nowadays, GDPR (cfr. Section 1.1.3) and other EU regulations (e.g., Convention 108+ for automatic data processing [34]) are currently in force to ensure data protection of European Union citizens. On the 10th of July in 2023, the European Commission adopted the adequacy decision for the EU-U.S. Data Privacy Framework [35], stating that the US guarantees the same level of data protection - as the EU - for data transferred from the EU to the U.S. The adequacy decision is a novelty introduced by GDPR that allows the transfer of personal data from the EU to foreign countries that meet an equivalent personal data protection level to

[2]https://privacylibrary.ccgnlud.org/case/maximillian-schrems-vs-data-protection-commission

[3]https://www.whistleblowers.org/news/the-case-of-edward-snowden/

that of the EU. Overall, the effort done by EU bodies is aimed at guiding organisations, companies, communities, and academics in pursuit of a common goal of guaranteeing data privacy.

1.1.3 General Data Protection Regulation (GDPR)

One of the most remarkable achievements in terms of privacy worldwide is the EU *General Data Protection Regulation* (GDPR) [36]. This law was enforced in 2018 and ensures strong guarantees about how companies must process personal data by respecting individuals' privacy. The seven pillars on which GDPR is based are the following:

Lawfulness, fairness and transparency. Personal data must be processed lawfully and transparently when the data is collected, processed, and during the process (e.g., after a cyberattack). In addition, those who collect data must guarantee that any discrimination towards individuals is avoided.

Purpose limitation. Companies and organisations must explicitly define the legitimate purposes for which they collect data.

Data minimisation. Companies and organisations can collect the strictly necessary data to accomplish a specific purpose.

Accuracy. The collected data must be accurate and up-to-date and then correct if needed.

Storage limitation. Companies and organisations can maintain data for the time required to fulfil the purpose. After this period, data must be deleted.

Integrity and confidentiality. Companies must introduce security measures to protect personal data from, for instance, unauthorised use, loss, destruction or damage. This requirement includes "appropriate technical and organisational measures," such as data encryption and access control.

Accountability. Companies and organisations must track their operations executed on personal data to demonstrate their compliance with GDPR.

Along with these fundamental principles, GDPR includes the most exhaustive definition of personal data that has been written so far in a regulation. This definition is incorporated in Article 4, which states the following:

"(...) 'personal data' means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;"

Thanks to this article, the notion of personal information encloses different data types - such as IP address, metadata, and online data - that are, thus, subjected to privacy guarantees provided by the GDPR. Another critical definition is the concept of "consent" of the data subject, who has to adhere to the processing of personal data. GDPR also ensures the right to be forgotten; namely, the data owners have the right to impose cancellation of their data to data controllers (e.g., an organisation or a company) and processors, but in some circumstances (e.g., for public health and interests).

In summary, GDPR is the most restrictive law in the world, protecting individuals' privacy of UE citizens. It imposes stringent obligations to both European and extra-European organisations that aim to collect and/or process data from people in the EU. Those listed above were some of the statements introduced by this law, and what is evident is the interest of the European institutions in determining solid guarantees of privacy for their citizens, starting with the definition of subjective factors (e.g., personal data), which may vary from one person to another.

1.2 Human Digital Twins

Human Digital Twins (HDTs) are an emerging technology originating from Digital Twins (DTs). DTs [37] are receiving growing attention both in the research community and the industrial sector, and are today considered a key enabler for boosting many application domains, such as, for instance, the healthcare one or Industry 4.0.

A digital twin is a digital copy of a physical asset that can be used for a variety of purposes, ranging from monitoring and predicting future status to optimising performances and preventing incidents. The first applications of DTs dated back to 1970, when NASA created mirrored systems monitoring physical objects (i.e., aircraft) to detect problems and, eventually, find solutions [37].

Since that, many researchers worked on the definition and improvement of this technology. As an effect of DT evolution and its benefits - e.g., real-time monitoring, predicting and making decisions affecting the physical part - this technology spreads in real application scenarios such as manufacturing, autonomous vehicular and smart grids.

Human Digital Twins (HDTs) are an evolution of DT, whose scope focuses on mirroring humans. Specifically, the human's virtual copy corresponds to a person, or a specific part, like organs or functions. HDTs can be used, for instance, in healthcare to find the best drug therapy, or in manufacturing to define the most suitable workspace in terms of ergonomics. However, a human is not only defined by physical aspects but also by intangible data defining the human beings (e.g., psychological state and environment) that must be modelled to define the virtual copy closest to the real one. Indeed, the more similar the virtual copy is to the physical entity, the more accurate the predictions made by the (H)DTs are.

Consequently, the quantity of data required to model an HDT is vast, and its type is strictly sensitive. These aspects make HDTs attractive to cybercriminals who aim to steal people's sensitive data. Hence, implementing cybersecurity mechanisms is essential for this technology, although they have only recently started to be considered. This thesis aims to provide privacy enforcement mechanisms to protect data used for modelling HDTs. In this section, we introduce this technology and illustrate the application scenarios.

1.2.1 From Digital Twins to Human Digital Twins

In a nutshell, DT technology aims to build digital models representing physical assets through mathematical models, specification-based techniques, application programming interfaces (APIs), and many other technologies, such as big data, artificial intelligence, machine learning, edge/fog/cloud computing, and IoT [38]. The main goal of a DT is to analyse data from the physical entities and make computations over them, predictions on the future states, and eventually, decisions to anticipate errors and relevant deviations of the physical system behaviour [39].

Contemporary to research advancements, DT technology has been applied in many real-world scenarios, such as emulating gas turbine maintenance and overhaul operations, or supporting process design decisions for trains' fleets [40]. Other examples of DT's application are in the field of anomaly detection for many different sectors, e.g., oil refineries [41], or cybersecurity. Many examples of real-world applications are also in the healthcare domain, such as the one developed by GE

Healthcare [42], which provides DTs for hospitals to support their capacity planning and improve their services. For instance, they provide support for bed and level of care configuration, surgical schedule and medical personnel management.

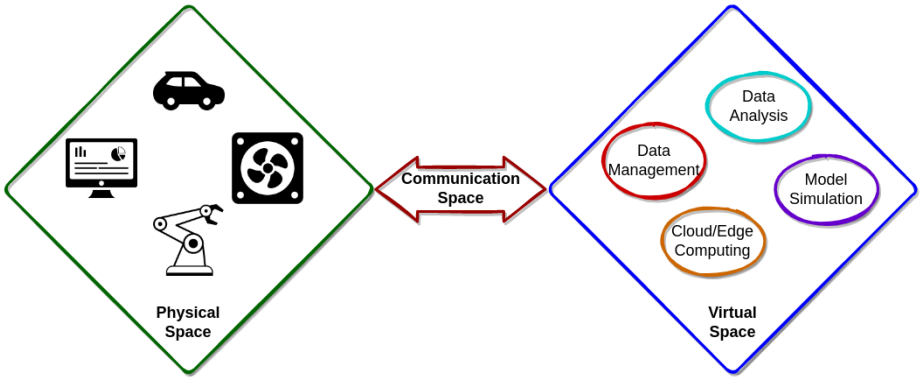


Figure 1.1: Digital Twin architecture

As depicted in Fig. 1.1, the digital twin architecture generally includes three main components [39], [43], [44], all supported by the cloud infrastructure: 1) the *physical space*, which groups all the physical assets composing the real entity; 2) the *virtual space*, which performs simulation processes over the data acquired from the physical space and data analysis tasks; and 3) the *communication space*, which is a bidirectional interface connecting the physical and digital spaces, allowing information exchange between them. The latter is essential as it allows the virtual space to be aware of the real environment. Specifically, the communication channel allows the virtual twin to conduct analysis, make decisions, and share the results with the real world. Multiple DTs can be combined to obtain a complete view of a system; for instance, an organisation may have a DT for each asset (e.g., each robot) interacting with each other to create a more complex model of the system [45]. This interconnection leads to DT networks allowing knowledge sharing and, consequently, improving modelling, monitoring, and prediction features. This cooperation is not limited to an organisation's boundary but can spread involving multiple stakeholders to create a more comprehensive system model [46].

DT technology is becoming the leading player in any large-scale intelligent system that requires real-time monitoring and interaction with the environment [47]. Many papers, such as [48]–[50], consider DT as one of the enabling technologies of Industry 4.0 and associate DTs with Cyber-Physical Systems (CPSs) for real-time data analysis, decision-making, and accurate execution of the production process. Further application domains are smart grid [51], and autonomous driving [52], [53].

Recently, Human Digital Twins (HDTs), a specific type of DTs emerged. An HDT corresponds to a person in the physical space modelled through data belonging to different categories [54], such as physical (e.g., biomechanics and anthropometric attributes), physiological (e.g., heart rate and muscle tension), perceptual performance (e.g., auditory and visual sensitivity), cognitive performance (e.g., skills and abilities), personality characteristics (e.g., propensity to trust or towards suspicion), emotional state (e.g., level of anxiety), ethical stance (e.g. values and beliefs), and behaviour. One of the main innovations introduced by HDTs is the potential to reproduce other aspects than physical and physiological characteristics. Indeed, an HDT aims to duplicate the human as an individual, including decisions, feelings, and social aspects. Although HDTs and DTs are similar concepts as they model a physical entity, share common technologies and require a communication channel, they differ in some key aspects. The first difference is that humans have mental activities that must be mirrored in the HDT, for instance, by combining physical information, such as respiration and blood pressure, detected by relying on biosensors. In addition, human reactions can be both subjective and objective. Moreover, humans are social creatures with ethics, and the environment influences them - e.g., the atmosphere, viruses, and air quality. Humans have decision power and the ability to employ products to achieve their goals [54]. Additionally, the synchronisation between the virtual and physical models is more demanding for HDTs than DTs, since human changes are more difficult to spot [55]. In addition, human data are heterogeneous as they represent different information and have different sources. Those differences make HDTs more challenging to design and manage than DTs.

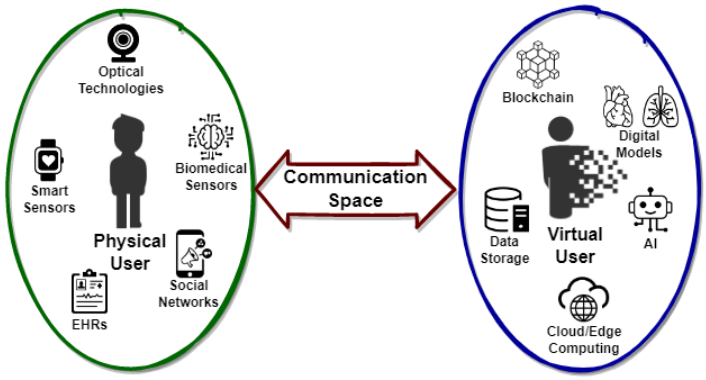


Figure 1.2: Human Digital Twin architecture

Fig. 1.2 shows a possible architecture of HDT. Here, we can observe that an

HDT represents human physiological and psychological by gathering data from different sources. For instance, optical technologies (e.g., cameras [56]) and body scanning systems can collect anthropometric data; haptic sensors can detect human intention; implantable biomedical sensors and electronic health records (e.g., Electroencephalography, pupil diameter, and breathing rate) acquire physiological data, which are also suitable to assess the human workload; finally, questionnaires and interviews support the collection of psychological data [57]. In addition to these technologies, wearable devices can be largely employed to sense data [58]. The virtual user simulates, in the virtual world, the physical counterpart in real-time through constant data synchronisation. It analyses the physical twin data to validate, optimise, evaluate, diagnose, give suggestions, and make predictions and decisions [1]. Such virtual replica includes data processing mechanisms, modelling technologies and data analysis and model optimisation [59]. The former encloses instruments for data cleaning (e.g., to solve data deletion), storage (e.g., databases and data management frameworks), and analysis. It also incorporates human modelling technologies which are required to simulate and create the virtual model mirroring the characteristics of the represented human, like AI and computing paradigms (e.g., cloud and edge computing). On the other side, data analysis and model optimisation are adopted to analyse the information and make decisions and suggestions. A further element is blockchain, as it allows to handling of data transparently and guarantees immutability [60].

1.2.2 Application Scenarios for HDTs

HTDs' usage is spreading in many application scenarios, such as for instance those aiming to improve healthcare systems, increase wellness and safety in factories, and provide humans with new features and entertainment modes. In this section, we analyse some of the most relevant HDT's application domains, providing use cases of this technology.

Healthcare

The application of HDTs in healthcare brings many benefits. The most relevant is the support for personalised medicine as an alternative to the "one-size-fits-all" method [61], where patients are treated with standard procedures without considering their habits, genes and environment. More in general, [62] refers to three main goals of HDTs in the medical field: preventive healthcare, medical healthcare, and communication between HDTs. In preventive healthcare, research is

devoted to creating silico patients, or specific organs, and systems to monitor their conditions and functions. The main goal of preventive healthcare is to gain personalised predictions and let doctors choose the best treatment for each patient's disease. In contrast, medical healthcare enables health institutions and organisations to provide smart health services and telemedicine. This application involves AI, Medical Micro Instruments, such as AR/VR and holograms, and patient data, like electronic health records, and IoT data. Finally, digital twins make possible real-time cooperation of the HDT with the environment, the physical patient, and other HDTs. This feature also promotes the cooperation of medical professionals, for instance, for patients with similar diseases, and simplifies the communication between patients and medical personnel [63].

The research on HDTs for healthcare is rapidly growing. For instance, the authors of [64] surveyed the exploitation of HDTs in aiding diagnosis, treatment and prognosis evaluation in cardiovascular diseases. [63] develops an HDT based on an ECG classifier to predict, monitor and treat heart diseases. This solution also allows healthcare professionals to collaborate, continuously monitor the health status of a patient, test the therapies in a safe environment, and prescribe the right care to the patient. Chakshu et al. exploited HDTs to estimate the blood pressure waveforms, finalised in detecting abdominal aortic aneurysms and their severity [65], and to predict carotid stenoses by modelling head vibrations [66].

Still, HDTs may improve diagnosis, treatment and management strategies for multiple sclerosis [67], and diagnosis visualisation and prediction in the oncology department [68]. The Virtual Brain (TVB) project,[4] which is part of the Human Brain Project supported by EPINOV,[5] leverages HDTs to define the best treatment strategy for epileptic patients, addressing the problem of uncertainty characterising epileptic therapies. Mainly, no reliable procedure exists to combine a patient's prognostic factors to predict the effect of surgery for the treatment of epileptic patients.

Other researchers focused on HDTs for therapy personalisation, as people do not always respond appropriately to drug treatment [69], [70]. To this aim, a research team from Linkoping University mapped RNA into an HDT to predict the effect of different types and doses of drugs.[6] The Swiss Federal Laboratories for Materials Science and Technology in collaboration with the University of Bern

[4] `https://www.thevirtualbrain.org`

[5] `https://ins-amu.fr/epinov`

[6] `https://liu.se/en/news-item/digital-tvillingar-hjalpmedel-for-skraddarsydd-medicinering-`

is exploiting HDTs to model the human body to control and predict the course of therapy.[7] Also, the EU project, EDITH[8] has the goal to build an inclusive ecosystem for HDTs in personalised healthcare. A further project is [71], which adopts Generative Artificial Intelligence (GAI) to improve personalised healthcare. This technology allows high-fidelity HDT modelling and supports the generation of synthetic data from statistical characteristics and patterns of the collected data.

The usage of HDTs in the healthcare domain is receiving growing interest also from industries with the development of many tools and products. For instance, FEops[9] and Philips[10] deliver HDT solutions supporting heart diseases detection. Also, Mai[11] technology allows modelling 2D body pictures in 3D to simplify the communication between clinicians and their patients and improve their education in therapy.

Physical Activities Coaching

Physical activity coaching is another application for HDTs. This domain encloses three main categories [72]: sports, well-being, and rehabilitation. In the first category, the digital replica represents an athlete, while in the second represents a person keeping a healthy life. In contrast, rehabilitation involves digital replicas of people who have lost their motor abilities. In any circumstance, the goal is to provide an appropriate coaching activity considering the human's current physical and mental condition. The coach, in turn, can exploit the HDT measurements and predictions to identify the best training sessions. For instance, [61] provides a fitness application of HDTs with which the coach can monitor the athlete's conditions and behaviour, aiming to increase their performance and health conditions. Instead, [73] exploits HDTs to create and readapt therapies and rehabilitation according to human factors, such as the psychological and cognitive state.

Manufacturing

Industry 4.0 paradigm integrates new technologies - e.g., Cyber-Physical Production Systems (CPPSs) - to guide modern manufacturing towards production process optimization. This evolution changes how human workers interact with the

[7] https://www.empa.ch/web/s604/eq71-digital-twin

[8] https://www.edith-csa.eu/

[9] https://www.feops.com/

[10] https://www.usa.philips.com/healthcare/resources/feature-detail/ultrasound-heartmodel

[11] https://www.mai.ai/digitwin/

system, defining the so-called Human Cyber-Physical Systems (H-CPSs). With this concept, the notion of Operators 4.0 arises [74]: workers collaborating with robots and machines in their tasks, resulting in a human-automation work system. In the human-automation paradigm, humans and machines collaborate to overcome each other's limits, where technologies enhance human capabilities and compensate for their limitations [75]. Human intervention is still fundamental in Industry 4.0. Even if processes are automated, thus guaranteeing efficiency, productivity, and cost savings, human workers' creativity and flexibility are essential in today's business, as those features cannot be automated [76]. Operators 4.0 thus require new qualifications and skills to handle the new industry technologies, and the task of teaching them is up to the factories. Consequently, the new industry paradigm requires introducing new technologies and supporting human workers in adapting and developing their tasks and HDTs are essential to achieve this ambition. Indeed, HDTs can estimate workers' conditions in real-time, considering stress and fatigue, and better identify the best training and ergonomics to increase motivation and safety. Thus, human characteristics, behaviours, and psycho-physical conditions must be included in the digital representation of the production systems, giving rise to the concept of Industry 5.0 [77]. HDT technology plays an essential role in achieving the integration of human workers in the new manufacturing paradigm by supporting their monitoring, production planning and scheduling, human-robot collaboration, and adaptive automation [78]. Human participation in manufacturing brings benefits, and the evolution of their skills will be reflected in the system's capabilities [79]. In addition, the worker's well-being contributes to the manufacturing process performance; thus, preventing the rise of phenomena like fatigue and stress is essential. For instance, [80]–[82] define an HDT framework to provide real-time feedback about production-line performance, allowing managers to modify and optimize the balancing or the assignment of the working tasks. Ergonomics is an additional element in manufacturing, since biomechanical overloads - such as wrong settings and postures, exerted forces, and repetitive actions - increase the risk of injury. To this aim, [83]–[86] define a framework for assessing the ergonomics indexes during the production processes. In addition, an HDT can simulate the production process also to support the evaluation of potential improvements through consecutive reconfigurations of the DT, such as in [86].

HDTs also involve human-robot collaboration, where robotic technologies support human workers in the most challenging workplaces - e.g., in heavy load manipulation and narrow spaces. For instance, [87] exploits HDT to manage workload

balancing, considering human and robot skills, human factors and the planning of the robot control program. Further examples are exoskeletons [88], that is, wearable robotics that human workers can adopt in manufacturing processes. Exoskeletons are used in many automotive industries, like FCA,[12] and IKEA,[13] which use arms, shoulders and legs supports. Other exoskeletons support workers in the logistics, construction, manufacturing, railway, aviation and marine industries.[14] A further proposal adopts HDT in the Human-Robot Collaboration (HRC) to handle uncertain situations, such as robot uncertainty (e.g., a robot runaway motion) [89]. In this event, the robot observes the HDT state (e.g., human position and intentions) and behaves accordingly, assessing human safety and task completion.

Another usage is enhancing human-machine collaboration processes to reduce the gap between the two parties caused by the fact that machines cannot replace humans in some intelligence-demanding tasks. In fact, humans bring creativity, decision-making capabilities, experiences, and intuition that a robot cannot replicate. To support people in these complex tasks, [90] proposes to use HDTs to extract workers' knowledge and imitate their information processes - i.e., their reasoning, behaviour, and actions while solving complex tasks - and use these to design intelligent technologies. In this way, machines can further support manufacturers by improving automation procedures.

Metaverse

The metaverse is a unified and perpetual virtual universe built from the union of technologies like augmented and virtual reality, artificial intelligence, and network protocols [91], [92]. Users can join this universe and interact with each other and with items using avatars representing their virtual identities. DT technology is fundamental in the metaverse to have a digital representation of real entities. Specifically, HDT can accurately model human identities, making the metaverse more attractive to people [93]. For instance, consider a company that exploits the metaverse for its professional meetings. This scenario requires virtual participants to resemble physical people as closely as possible to have a rich and immersive experience. HDT is fundamental in this scenario, as it can be used to define the metaverse identity, which may be associated with a 3D modelling of the physical

[12] https://www.media.stellantis.com/br-en/corporate-communications/press/fca-tests-exoskeleton-in-the-production-line

[13] https://germanbionic.com/en/ikea-designs-workplaces-with-cray-x/

[14] https://www.sarcos.com/; https://ottobockexoskeletons.com/?lang=en; https://eksobionics.com/

person. However, it is not limited to modelling a general physical aspect; in fact, this identity encompasses sensible data such as health and behavioural data [94]. Currently, the market offers several solutions for meeting in the metaverse, like Microsoft Mesh,[15] AltspaceVR,[16] and MeetingVR.[17] In addition, the metaverse is a platform for new virtual games and entertainments, like Sandbox,[18] Roblox,[19] and Decentraland.[20] People join those virtual worlds to live a parallel life where they can meet other people, participate in events - like summits, forums and concerts - and have experiences. For instance, during the Metaverse Fashion Week in Decentraland,[21] users could wear virtual garments and purchase them, receiving a physical version. Moreover, many brands are expanding in the metaverse, creating their virtual worlds, like Nikeland,[22] and Gucci Town[23] just to mention few.

The metaverse is an ever-expanding world, and, according to Gartner, 25% of the world population will spend at least one hour per day in this virtual universe by 2026 [95]. This trend highlights that people will join the metaverse for more activities, including the more personal and not entertainment-related ones - e.g., meetings, business, and purchases. Consequently, HDT will become more relevant in gaining an accurate representation of humans.

[15] https://www.microsoft.com/en-us/microsoft-teams/microsoft-mesh
[16] https://www.meta.com/en-us/experiences/pcvr/altspacevr/1072303152793390/
[17] https://www.meetinvr.com/
[18] https://www.sandbox.game/en/
[19] https://www.roblox.com/
[20] https://decentraland.org/
[21] https://mvfw.org/
[22] https://www.roblox.com/games/7462526249/NIKELAND-ZOOM-FREAK-4#!/about
[23] https://vault.gucci.com/it-IT/story/metaverse

2

Security and Privacy Issues on (H)DTs

On the one hand, DTs and HDTs introduce many benefits in several application scenarios (cfr. Section 1.2) and, because of that, their usage is rapidly increasing. On the other hand, their characteristics make them an attractive target for cybercriminals.

One of the essential features of DTs, which is also the main reason for their success, is the high fidelity with which the virtual twin represents the physical system. This is one of the main requirements to build an accurate virtual model that is as similar as possible to reality. Fidelity requires continuous synchronization between the real and virtual systems, which minimizes any discrepancy through constant interactions. However, this requirement raises the issue that any malicious update to the virtual world leads to a deviation from reality which, eventually, affects the physical system [48]. Indeed, if a digital twin receives incorrect data, it will make wrong interpretations, conclusions and decisions, sending an inappropriate answer to the physical system and damaging its correct behaviour.

A further consequence of fidelity is that the virtual world contains all data and information of the physical entity, and such data might contain sensitive or private information. For this reason, cyberattacks may lead to severe consequences, like leakage of confidential and private information, damage to reputation and financial losses [43], [96]. In addition, the DT can become a blueprint for the real system, as

it identifies the components' behaviours and interfaces [96]. As a result, attackers can exploit the digital twin to acquire more information about the physical twin and refine the attack strategy.

The interconnectivity between DTs and other systems is another aspect that increases the attack vectors, with a consequent raising of potential security issues [97]. In this regard, some characteristics of DTs, such as the implementation of heterogeneous technologies, decentralised structure, and continuous data synchronisation, this technology requires more than the conventional approaches to address privacy and security issues [98].

The first contribution of this thesis [8] is to analyse the security and privacy issues affecting DT and HDT technologies in Section 2.1, then it lists the security and privacy challenges introduced by HDTs in Section 2.2.

2.1 Security Issues on DTs

Cybersecurity issues in DTs have started to be investigated only recently by identifying the leading causes of security weaknesses. The same issues also affect HDTs as they are a specific case of DTs and thus are based on the same technologies and principles. However, HDTs introduce further issues (cfr. Section 2.2) addresses. This section considers the three components characterizing the DTs - i.e., physical, virtual and communication spaces (cfr. Fig. 1.1) - and analyses the related cybersecurity threats and possible countermeasures, based on the current research findings. Table 2.1 summarises these threats.

2.1.1 Threats Against the Physical Space

The physical entity mirrored by the DT may be composed of a unique physical component - e.g., the engine of a car - or a set of connected components - e.g., a whole car with its sensors, onboard computer, steering wheel, etc. In the latter situation, the physical asset is, in turn, organized in three layers: 1) the physical components (e.g., sensors); 2) the software, eventually managing the physical components; and 3) the communication channel that components use to exchange data. Consequently, securing all the underlying infrastructure is crucial as vulnerabilities can affect hardware, software and communication networks [97].

Starting from the first layer, i.e., sensors; these are devices acquiring data from the environment, such as IoT devices, and are subject to different well-known security weaknesses. For instance, IoT security gaps are mainly due to the limited resources of IoT sensors, making security settings hard to implement. Moreover,

Table 2.1: Threats affecting the singular DTs spaces

Space	Involved Asset	Threats
Physical	Sensors	Weak security settings
Physical	Communication channel between sensors	Network attacks, Illicit access, Ransomware
Physical	Software	Privilege escalation
Physical	Communication channel	Network attacks, spoofing, DoS, MitM, eavesdropping
Virtual	Computing infrastructure	Known security issues, DoS
Virtual	Software - e.g., databases and applications	Known security issues, full access rights to software maintainers, living of the land attacks, reverse engineering
Virtual	Virtualisation systems	Privilege escalation, introduction of rogue servers, Dos
Virtual	Machine learning algorithms	Injection, poisoning, evasion, impersonation, inversion attacks
Communication	Standard network protocols	Known security issues, network flooding, DoS, illicit access, operational disruption
Communication	Data transmission	Data tampering, Man-in-the-Middle, eavesdropping, Sybil attacks

many threats come from the communication channels used in the information exchange process [99]. Hence, sensors are usually victims of several network attacks, unauthorized accesses, and ransomware.

Software inside the physical components can be accessed through privilege escalation [39], where adversaries acquire security parameters or credentials through network attacks. It is worth mentioning that human activities, voluntarily or not,

pose significant security challenges in the system [100]. In particular, social engineering attacks consist of manipulating a victim to access sensitive information or gain money, avoiding any security measure [101]. There are many ways to perform this kind of attack and the most famous is phishing. For instance, an employee may receive an email requesting to open a URL to update the username and password. However, behind the website, there is an intruder who steals login data to access the system and compromise it. Consequently, attackers may exploit unauthorised privileges to disconnect the devices in the physical layer, change configurations, generate false values or manipulate the network traffic.

Finally, components of a physical asset are connected through the network and thus could be victims of attacks exploiting that channel. Network attacks are, for instance, Denial of Service (DoS), Man in the Middle (MitM), eavesdropping, and spoofing [39], [43].

2.1.2 Threats Against the Virtual Space

Generally, the virtual space can be split into the virtual representation of the physical twin and all the tools to acquire knowledge and make predictions. The former encompasses all the technologies required to represent, mirror, and monitor the digital world's physical counterpart. Instead, the latter groups all the tools (e.g., artificial intelligence algorithms) that generate knowledge and predict possible behaviours by exploiting the data acquired from the physical world. The damage caused by attacks against this space depends on the purpose of the DT. For instance, if a DT is designed for intrusion detection [102], attackers can send malicious data going unnoticed. Alternatively, in the case of a DT for anomaly detection [103], attackers can compromise the actions performed by the virtual twin to detect anomalies by, for instance, sending fake data, directly modifying the source code or inputting harm code. This attack may cause the DT to misjudgment, thus letting hackers act unnoticed.

Additionally, DTs usually use computing infrastructure (i.e., cloud/fog/edge) to process and store the data acquired from the physical world. Those technologies suffer from well-known security and privacy issues, and an attacker can tamper with them to compromise data [39], [43]. Those resources can also be victims of DoS attacks if an attacker overworks them by requesting additional resources, causing communication, computation and storage overload [39].

The software components represent a further threat against the virtual representation, including DBMSs and applications [39]. Usually, any software contains

known failure points that hackers can exploit in their activities, like stealing - threatening data confidentiality, allowing illicit acquisition of sensitive information [104] - or modifying the data. In addition, organisations purchase software deployed and maintained by third parties that may exploit their full access rights to steal private information [39]. Attackers can also perform a "living off the land" attack by abusing benign tools used by the user to perform legitimate tasks aiming to compromise the DT behaviour [105]. Moreover, poor coding makes software components weak to reverse engineering attacks.

As in the physical world [39], adversaries can escalate privileges to gain access rights to the virtualisation system, VM or container and exploit them to attack other virtual resources. This can be performed by an intruder or by an attacker who successfully perpetrated a social engineering attack. Intruders can also introduce rogue servers and infrastructures to take control of the digital twin. Moreover, they can request additional resources from the virtual components to overcharge them and cause DoS.

The other component of the virtual space consists of the tools required to extract, model, and analyse data acquired from physical space to obtain meaningful information and build an adequate virtual model. These tools include machine learning algorithms that have known security gaps. For instance, hackers can compromise the training phase through a poisoning attack, i.e., by injecting artificial data into the training data set, aiming to reduce the models' accuracy. Also, an attacker may modify the trained gradients or the model parameters aiming at deteriorating the knowledge inference performances [98]. Generally, machine learning algorithms are potential victims of poisoning, evasion, impersonation, and inversion attacks that corrupt the final model [43].

2.1.3 Threats Against the Communication Space

The digital and the physical spaces exploit a communication channel to exchange data acquired in the physical environment - from the sensor to the digital twin - and the decisions made by the DT - from the virtual to the physical asset. This channel usually exploits standard protocols with well-known vulnerabilities, which hackers can exploit to analyse and redirect traffic or to modify the shared data [43]. In addition, attackers may flood the network traffic or send incorrect data to crash the system, causing a DoS attack [43].

The digital and the physical spaces exploit a communication channel to exchange data acquired in the physical environment - from the sensor to the digital

twin - and the decisions made by the DT - from the virtual to the physical asset. A big issue is data tampering attacks, which can modify, replace, forge, or remove data exchanged through this channel, leading to false knowledge and inconsistent DTs [98]. In fact, this channel usually exploits standard protocols with well-known vulnerabilities, which hackers can exploit to block communications, cause operational disruption, and access data without authorisation [100]. In addition, attackers may flood the network traffic or send incorrect data to crash the system, causing a DoS attack [43]. Further weaknesses of the communication channel are Man-in-the-Middle attacks - during which an adversary intercepts and alters the exchanged data -, eavesdropping - when an intruder accesses data during its transmission -, and Sybil attacks - in a decentralised system, this attack happens when a node simultaneously manipulates many identities to gain influence and undermine the service's reputation system [98].

2.2 Security and Privacy Challenges for HDTs

HDTs suffer from the cybersecurity issues listed in Section 2.1. However, they require addressing additional challenges with respect to the ones that characterise all DTs. Indeed, HDTs are frequently employed in highly sensitive domains, like healthcare (cfr. Section 1.2.2). These domains, in turn, require processing sensitive personal data, such as those related to a specific organ health situation. Since the modelling of a high-fidelity HDT requires a high amount of data, which is not always available or may be hard to acquire, adopting different collection methods as in [71] may be required. In addition, due to the sensitivity and the amount of personal data required for the HDT to be generated and maintained, their protection from attackers is crucial. For instance, an attack on an HDT can endanger the life of the patient to whom the HTD belongs. An attacker may directly target the patient's health by leading the system to supply the wrong drug quantity by manipulating the HDT [106]. Another threat is biometrics hijacking,[1] as data in HDTs can be used for identity theft and other types of fraudulent activities.

In what follows, we discuss some of the main open research issues, focusing on those related to privacy.

[1] https://medium.com/humancyberhub/cyber-security-of-human-digital-twins-5808ffbad28b

2.2.1 Privacy Preferences

The primary goal of HDTs is to supply customised services for users leveraging on the acquisition, in most cases, of highly sensitive personal data. This raises the need to make the user able to specify privacy preferences and have them enforced by the system. In particular, when users share their data with a service provider, they should be allowed to manage different factors, including what data type to share, for which purpose, with whom, and for how long data can be retained. All these aspects influence the users' reservations when disclosing their data. For instance, they can make different choices when personal data are shared with a hospital than with an e-commerce platform. Letting users define their privacy preferences permits them to manage all the above requirements.

Thereafter, privacy preferences should target both HDT usage, as well as the data that are employed to build the HDT. Another important aspect to consider is the context in which users employ their HDTs. For instance, a manufacturing service may be authorised to access HDT data only when the person to whom the HDT belongs is on the company premises and during working hours. On the contrary, the service can not access the HDT if the person is at home, even during working hours.

While other components of privacy preferences can be statically defined (e.g., which data to share, with whom, and the retention period), the context is more dynamic. Indeed, it models conditions, which may vary, such as the location, time and type of activity in which the service can access the data.

2.2.2 Multiple HDTs

A further issue is that HDT applications are usually developed for a specific use case. The result is that a user may have multiple HDTs, providing their representation at different levels of detail. For instance, a user may hold an HDT built for testing a specific medical treatment and one for interacting in the metaverse or doing fitness exercises. Following this trend, in the near future, a person will have multiple digital twins to manage. Moreover, the granularity of HDT models can be even finer. For instance, instead of having a single HDT for all healthcare services, a person may have one HDT for drug therapy, one for heart disease and one for physiotherapy. This would require maintaining a high number of HDTs, raising data management issues, possible inefficiency, and difficulties in cooperation among applications. In addition, further effort is needed if a situation requires joining some of the virtual models.

Focusing on privacy issues, holding multiple HDTs magnifies the need to enforce proper security mechanisms, which, on the other hand, brings further privacy threats. First of all, the management and enforcement of privacy preferences on HDT usage might become more complex. Additionally, it is hard to determine the privacy risks and possible inferences that may happen from accessing different HDTs related to the same person.

A promising approach to managing multiple HDTs is to leverage a unified HDT model [54]. According to this view, each person will have a unique digital model retaining all his/her data (physical, physiological, environmental, etc.), which each application can access to provide the requested service and build the needed HDTs. This approach permits the same model (or, better, a view of it) to be used in many different application domains.

This paradigm also facilitates the control of HDT usage through privacy preferences since preferences can be directly specified on the unified HDT. Therefore, even if the unified model retains all user's personal data, each service gets only those needed to supply its specific task (e.g., a training service does not need to access detailed information about the person's organs).

However, even if services obtain partial data related to different contexts (e.g., workplace, gym and hospital), they can obtain more data than the allowed ones by making inferences (e.g., joining two views of the same HDT). Thus, proper privacy-preserving mechanisms - such as anonymisation techniques - are fundamental.

The definition of those mechanisms must consider the possibility of defining different levels of data release to the various services and, therefore, different levels of details according to which the resulting HDT models its physical twin. Moreover, there is the need to properly protect and store the data belonging to the unified HDT model (for instance, those related to organs, the nervous system, and the psychological behaviour of a user). In this respect, a promising approach is a convergence between the unified HDT model and Personal Data Storage (PDS) [107]. PDS allows a shift from a service-centric to a user-centric data storage architecture by offering individuals the capability to keep their data in a unique logical repository that can then be shared with third parties under the control of end users.

2.2.3 Interactions Among HDTs

Further privacy issues may arise from the interactions among HDTs belonging to different persons. Those issues are, for instance, crucial for metaverse applications,

where users directly interact through their HDTs. A first relevant issue is how to make users able to easily define which data should be shown to other users with which they interact. A further challenge is how to manage conflicting privacy/security requirements that different users may have. For instance, a person may want to access a virtual world with a different physical aspect not to be recognised. On the contrary, another user may not be willing to interact with such kinds of HDTs, since he/she considers this a threat to his/her security. Thus, collaboration among different HDTs requires the enforcement of multiple privacy preferences and finding a balance between the quality of the delivered services and the privacy/security of the different users to which the collaborating HDTs belong. Still related to the metaverse, another issue is about the representation of a deceased person (e.g., an artist). In fact, this practice - a "digital resurrection" - may violate the dignity of the represented individual [59]. Therefore, balancing privacy/security measures with the quality of the service must be combined with ethical considerations to build a virtual environment where different HDTs can harmoniously coexist.

2.2.4 Ethical Aspects

The sensibility of data involved in HDTs raises ethical issues. For instance, HDTs might not be accessible to anyone, increasing diversity between those who can access more personalised services and those who cannot [108]. Consequently, to cite an example, there would be people who, thanks to HDTs, will benefit from better medical care. An additional problem arises if HDT data are used to build AI models, for healthcare or other application scenarios. Here, there is a need to ensure that, even if HDTs are not broadly accessible, their data do not create biased and unfair models. On the other hand, HDTs might expose different types of data about individuals (e.g., health status, personality features, emotional state). When a service gets access to this data, we must ensure that it is not exploited for unethical purposes. For instance, we have to avoid that an employer may exploit the accessible health information on an employee to create unfavourable conditions and force him/her to resign. Further issues are related to the morality behind using HDTs to enhance human capability [108]. For instance, usually, athletes work hard to increase their performance; what if they can achieve the same results thanks to HDTs?

Finally, does the HDT data would follow the owner's life? For instance, [109] questioned people aged 60+ about the destiny of their HDTs after their life ends. While most of them preferred that their HDT would be deleted when they died,

others would agree to donate the anonymised version of their HDT for medical purposes.

2.2.5 Approaching the Challenges

In what follows, this thesis discusses how some of the security and privacy threats and requirements presented in this chapter can be tackled. Specifically, Chapter 4 illustrates a framework enabling users to specify and enforce their privacy preferences for personal data sharing across different contexts. In contrast, Chapter 5 provides a solution that securely lets users use their personal HDT with privacy guarantees.

3
Related Work

The main research topic of this thesis centres on HDTs and, more specifically, on ensuring privacy for users who use this technology as a means of accessing services and sharing their data with providers. This section first shows the existing security and privacy enforcement mechanisms the literature proposes to face with the security and privacy challenges for DTs and HDTs presented in Section 2.1. To better catalogue the security/privacy mechanisms, Section 3.1 divides them into the three layers composing the (H)DT: virtual, physical and connection spaces. Subsequently, Section 3.2 analyses the state of the art related to policy negotiation when multiple users and service providers interact in the scope of a service. Finally, since we started by studying the enforcement of privacy preferences in the edge computing environment - Section 3.3 lists the state-of-the-art proposals defined for this paradigm.

3.1 (H)DT Security Mechanisms

DTs and HDTs are built on the same technologies, such as sensors, network protocols and databases; therefore, they share the same security threats. Consequently, the countermeasures to protect the two paradigms are similar, depending on the specific component. In this section, we illustrate the security mechanisms proposed

by the literature for each layer composing the (H)DT, namely physical, virtual and connection spaces.

3.1.1 Pysical Space Countermeasures

In general, the physical space incorporates both Information Technology (IT) components as well as Operational Technology (OT) components, including IoT devices and Cyber-Physical Systems (CPSs). Since security in OT mainly involves industry physical assets, the goal is to protect their safety, productivity, and reliability on all fronts: cyber, physical and social dimensions [100]. Moreover, solutions to protect the physical components are related to the specific asset type. For instance, to protect IoT devices, [99] identifies several possible countermeasures, namely, authentication, lightweight encryption, anonymisation, firewalls, intrusion detection and prevention systems (IDS/IPS) and risk assessment. An example of IDS is proposed in [110], which is based on a hybrid model of deep learning and random forest to detect different types of network attacks. While deep learning is used to extract important features from network data, the random forest is used as an attack classifier. Recent generations of IoT devices have included advanced security features like the Hardware Security Module - a tamper-resistant physical entity that improves the protection of cryptographic keys, encryption data exchange, and other data types - and Trusted Execution Environment - an isolated environment where computations are running ensuring confidentiality and integrity [111]. Furthermore, authentication, is crucial in providing security, privacy and confidentiality, and it can be enforced through identity-based authentication techniques (e.g., hash, symmetric, or asymmetric cryptographic algorithms) [111].

Also, network attacks against the communication channels must be limited to prevent attackers from obtaining security parameters that allow them to access the physical system. Some solutions are privilege separation, monitoring and network isolation through, for instance, intrusion detection/prevention systems, firewalls, and, also, good practices [39]. In addition, the authors of [43] also suggest deploying Software-Defined Networks (SDN) to protect telecommunication networks.

3.1.2 Virtual Space Countermeasures

Possible countermeasures to protect the virtual representation space are privilege separation, network isolation, and monitoring [39], applying good practices in software development, such as secure coding methodology, peer code review, and

testing since the earliest developing steps [39], [43], [48], [96]. Other mechanisms that can be applied are software hardening, active defence techniques, and copy protection[96].

Moreover, the solutions proposed so far to protect data from unauthorised modifications exploit tamper-proof and tamper-resistant hardware, blockchain, and hash functions, as, for instances, in [43], [105]. Another important element is data integrity in the creation and maintainment of consistent DTs. To this aim, [112] proposed a blockchain-based Provable Data Possession (PDP) as a synchronised lightweight method to assess the integrity of data blocks and verify time states in virtual spaces.

Recently, the controlled sharing of (H)DT data has started to be investigated. In this envision, access control mechanisms are relevant to protect resources, such as databases, containers and VMs, from unauthorised access. These proposals favour the paradigm of the unified HDT model, which minimises the need for generating multiple HDTs for an individual (cfr. Section 2.2.2). Managing many HDTs for a user can be burdensome and potentially lead to privacy issues due to the complexity of their management. On the other hand, having a single HDT might also raise privacy issues because service providers could gain access to all the personal data. Access control mechanisms can be employed to address this challenge because they restrict access to a specific portion of the HDT. Therefore, individuals would have a single HDT, and service providers would have access only to the relevant data. Among the proposed solutions, [113] proposes a tag-based access control mechanism, where accesses are controlled by tags attached to DT data. Also, [114] proposes a blockchain-based solution for HDTs to generate instances accessible to care providers.

Adopting blockchain is a valuable solution for dealing with Cloud, Fog, and Edge computing issues. For instance, [115] replaced cloud/for computing with a revisioned version of the blockchain, called twinchain, which is resistant to quantum attacks, as it adopts a quantum-resilient cryptographic scheme. In addition, it reduces transaction confirmation time, thanks to the ad hoc mining process that can boost the block creation procedure. [116] proposes a consensus mechanism to support transactions between Intelligent Transportation Systems (ITS) and DT services (e.g., edge and cloud servers). The objectives are twofold: the first is to provide an on-demand DT as a service architecture for ITS by adopting DT services, and the second is to reduce the time required to generate transactions by designing a DT Delegated Proof of Stake consensus protocol.

The protection of AI algorithms from being compromised is crucial to ensure

fair HDT models (cfr. Section 2.2.4). This protection, in turn, translates to having reliable copies of individuals, which are vital for being employed in sensitive applications where precise data is essential to guarantee correct services. In such cases, if the data is compromised, the safety of the individual could be seriously threatened - such as in healthcare contexts. Among the solutions offered by the literature to cope with attacks against AI algorithms, there are data sanitisation, security assessment mechanisms, privacy-preserving techniques and, in general, the design of secure learning algorithms [117]. A further method to address common issues of machine learning consists of adopting federated learning [118]. An example is [119], where the authors use federated continuous learning to solve the problem of privacy leakage in machine learning training when data is outsourced to the server. Thanks to this mechanism, the proposed solution allows to safe use of real-time data, as clients do not have to retrain when new data is generated.

3.1.3 Connection Space Countermeasures

Protecting the connection between the virtual and physical space is crucial to avoid data leakage and tampering during the transmission from the physical to the virtual space. To this aim, it is essential to guarantee data integrity and confidentiality, and Cryptography and security protocols are some of the techniques that may enhance the security of the communication channels from this point of view [39], [43]. This protection is also essential to protect people from being harassed when their HDT is stolen, preventing illicit and unethical data use (cfr. Section 2.2.4). Different solutions exploit SDN, redundant network paths, and other innovative means - such as blockchain and smart contracts [43]. Some proposals using blockchain are [120], [121], which guarantee data integrity by combining blockchain technology with a distributed hash table (DHT). While the former stores metadata and references to the collected data, the latter stores the sensed data. Confidentiality is ensured through data encryption before sharing data, like in [120]–[122]. Also, [123] defines a Revocable Attribute-Based Encryption to limit data access only to authorised parties. Another proposal is [124], which uses Physical Unclonable Functions (PUFs) to protect data transfer between the physical and virtual spaces: a hardware-bassed security mechanism implementing electronic circuits used for identification, authentication and secure communication purposes. In this solution, the authors used a PUF-based protocol to identify each device and perform authentication. Moreover, data are encrypted while shared through the communication channel.

Finally, DoS attacks can be mitigated by deploying the DT system in a decentralised architecture, such as fog and edge computing [39].

3.2 Policies Negotiation

The identification of a representative policy for a task shares some similarities with the problem of policy negotiation in multi-user scenarios. Both require the selection of the most appropriate policy based on a variety of factors and priorities. Notable examples of policy negotiations for multi-user scenarios are those defined for social networks, where various approaches have been proposed to address content-sharing conflicts, that is, when a resource, like a picture, is tagged with multiple users with different sharing preferences (e.g., [125], [126], [127]). Despite these solutions negotiate an optimal sharing policy, they are limited to access control policies, whereas our proposal focuses on privacy policies and user privacy preferences.

In the case of interactions among HDTs, one of the main issues arises when users have different privacy preferences (cfr. Section 2.2.3). In this matter, negotiation can cope with this challenge since it identifies intermediate privacy requirements. This procedure becomes fundamental to finding agreement between parties and facilitating their interaction, especially in environments that involve many people participating, such as the one of HDT applications. In the field of privacy management, we can mention [128] that introduces a privacy-preserving negotiation mechanism to generate a candidate policy from two sets of privacy preferences based on a boolean circuit and homomorphic cryptosystems. However, the negotiation considers only data type and not other privacy aspects like purposes, third parties, and retention periods. [129] describes privacy preferences and policies through an ontology, including the field "allowsNegotiation", stating the possibility of policy negotiation in the case of conflict. However, this allows only the selection of either the user's or the provider's conditions, unlike our approach, which seeks an intermediate policy that balances all participants' needs. [130] proposed a three-stage negotiation protocol that aligns a server's data handling policies with a client's privacy preferences. The three-stage process starts with the server making an initial proposal to the client by sharing the required and preferred conditions. If the requirements are fewer than the client's ideal conditions, the client accepts the policy; otherwise, it rejects it by making a counter-proposal based on its privacy preferences. The server then evaluates this counter-proposal

and formulates a ”best offer” by integrating required and preferred conditions. The proposal in [130] differs from our proposal as it is a bi-directional negotiation protocol, while our proposal aims to find a policy agreed upon by the multiple involved parties.

The closest work related to ours is the one proposed in [131], which defines a policy negotiation mechanism for an IoT scenario, where IoT owners/servers manage the negotiation and access requests to the IoT infrastructure (corresponding to our service providers), and IoT users either access data or provide personal information. Both parties define their privacy preferences or requirements, which are analyzed during the negotiation through a utility-privacy trade-off function that balances the benefits of data exchange, privacy exposure, and privacy sensitivity perception. The protocol offers a group negotiation procedure that merges all users' privacy preferences to generate a new, most conservative policy, called the ”boundary policy”, which ensures no user's privacy preference of group members is violated. This boundary policy is employed to define the aggregate (i.e., representative) policy during negotiation. The aggregate policy is selected between all possible combinations within the boundary policy using an optimization problem with two objective functions, aiming to minimize privacy exposure and maximize service utility. In the first negotiation round, the optimization focuses on policies that comply with the boundary policy, while in the second round, the system asks one or more users if they are willing to compromise their privacy for a better service. If the last phase fails, users are notified that their privacy requirements cannot be met. The main difference with our proposal relies on the negotiation protocol favouring IoT users. In fact, this negotiated policy considers the more conservative preferences, and the optimization problem minimizes the data collected from the users and maximizes their benefits. Instead, we aim to provide an optimal solution for both users and service providers. In addition, this negotiation considers the existence of one IoT owner/server; meanwhile, our solution is conceived for multiple service providers.

3.3 Edge Computing

Several solutions for protecting users' privacy at the edge [132] are available in the literature. Some of them aim to support edge computation on user data while protecting the user's identity, location, or/and data itself.

Starting from identity protection, one of the principal security requirements

related to edge computing is ensuring *anonymity* when users authenticate to edge nodes [133]. During this process, users' sensitive data passes through insecure communication channels, and an attacker may be listening on these channels, stealing personal information. The present research offers different solutions to address this issue, some of which rely on the definition of pseudo-identities. One of these proposals is [134], where users share their pseudo-identities when they need to connect with edge nodes. Each pseudo-identity has a time duration and is a combination of the real user's identity with a random number, so an attacker cannot do reverse operations to trace back to the user's identity. [133] generates a fixed pseudonym for each user based on the password and biometrics. Such a pseudonym is then combined with a one-time password to define a one-time pseudonym used for authentication. Also, in [135], users' identities are passed to a hash function to hide them. Other solutions ensure anonymity through the combination of pseudo-identifiers with blockchain. For instance, [136] stores metadata for authentication and tracking on the blockchain. Metadata is indexed by pseudonyms, which derive from the real users' identity and a random number. In addition, pseudonyms are masked by random values during authentication. Another proposal is [137], which adopts pseudonyms and a hierarchical blockchain architecture for their management, allowing efficient pseudonym management. The first blockchain layer is a main chain deployed on the cloud, recording and verifying pseudonyms. Edge nodes maintain multiple subchains, working as miners to add pseudonyms. Finally, a relay chain handles the requests between the main chain and the subchains.

Users' *location* is another significant problem for edge computing on different occasions. For example, an untrusted entity may track the users' position by identifying the edge node with which they interact and the service they are using. Also, task offloading procedures can jeopardise the users' location privacy. A solution consists of achieving k-anonymity relying on the Location Trusted Service (LTS) as an intermediary for location-based queries, allowing a provider to receive an area instead of the precise users' position [138]. Specifically, this paper implements a hierarchical distributed LTS, where LTSs are distributed among the edge nodes to create small territorial competence zones (0-zones). The adjacent 0-zones are clustered together to create a new layer (1-zone). Then, 1-zones are clustered to create a new layer, and so on, until a forest of trees is created. Thanks to this architecture, zones in the highest position in the hierarchy only know the number of users positioned in each sub-zone but without knowing the exact location. Another location protection technique consists of adding noise at the Radio Access Network level to add confusion and prevent an attacker from identifying

the user's position [139]. [140] defines a proxy application as an intermediate node between the user's device and the edge node. The communication between the proxy and the user's device is tunnelled; thus, the provider would only know the proxy's position. Other solutions focus on protecting the users' identities, preventing them from being tracked. For instance, [137] frequently updates the users' pseudonyms, and [141] adopts deniable authentication, which allows an edge node to authenticate a user's device without proving the existence of this connection. Another proposal operates on the delocalisation of services on edge nodes by balancing privacy and utility [142], so that users do not have to reveal their location. Still, other researchers leverage differential privacy, like [143], where users' devices generate some false location data before sharing this data. Also, in [144], the edge nodes generate an obfuscation range used by the user's device to generate an obfuscated address. [145] defines an offloading strategy which selects a region based on the user's location, privacy requirements, and task size. Then, the user generates a perturbated location based on the determined region. [146] safeguards the user's location privacy by adding noise to the offloading ratio based on the channel conditions and privacy needs. Finally, [147] replaces real locations with virtual locations to which it applies random noise to perturb data.

Lastly, *data protection* is essential to prevent the leakage of sensitive personal data. Generally, it is ensured through data encryption before its sharing/storage, like [148], [149]. A further proposal is [150], which deploys an architecture for offloading to decide whether to compute a task on the user device, edge node or cloud server. This decision depends on the computation of a privacy sensitivity level, which includes integrity, confidentiality, availability, and real-time accessibility at different times and locations.

Other solutions aim to regulate data usage on the edge level, such as ensuring that only authorised entities can access/elaborate users' data. Here, different access control models have been proposed, and one of the most famous is the Role-Based Access control (e.g., [151], [152]) Also, [153]–[155] propose different crypto-based enforcement through Ciphertext-Policy Attribute-Based Encryption. Other projects are [156], [157], which extend the Attribute-Based Keywords Search (ABKS), and [158], which develops a lightweight privacy-preserving cyphertext retrieval scheme extending Cyphertext-Policy ABKS (CP-ABKS). Another technology employed for access control is blockchain. In [148], [149], [159], proposing a data-sharing scheme which relies on smart contracts to validate access and grant permissions for data requests.

Among these, to the best of our knowledge, only [151] considers the context as

we do in this thesis. Specifically, [151] uses RBAC and contexts to enforce privacy policy compliance, i.e., to ensure that data usage adheres to the privacy policy specified by the data consumer. Although these solutions are useful, our proposal takes a different approach to user privacy by allowing individuals to express their preferences regarding the usage of their data at the edge level. This work can be extended to HDT-based applications, letting users specify their privacy preferences, stating who is authorised to access their data, for what purpose, with whom this data is shared, and for how long it is retained, in different contexts. This solution addresses the need for defining privacy preferences, considering the context, in environments where users' privacy can be easily jeopardised (cfr. Section 2.2.1).

4

Context-based Privacy Preferences Enforcement

Privacy-based data acquisition is essential in HDT modelling and requires defining and enforcing privacy preferences, which should incorporate the user's specific context. In fact, privacy preferences may vary depending on, for instance, where the user is, what time or which day it is, if the user is alone or with someone else, and so on. To address these complex privacy requirements, we developed ConPrEF [10], a novel privacy enforcement framework that allows users to define their privacy preferences for data collection in HDT modelling based on a rich set of contextual features, such as location (e.g., a user enables the collection of health data only if in a hospital), time (e.g., a user agrees to collect the stress levels only on working hours), activity (e.g., a user shares the heartbeat with the edge node only during the training), and situation (e.g., an emergency may force a complete data sharing of health data for medical intervention). Additionally, ConPrEF allows constraining data sharing based on other individuals' presence in a given context (e.g., the user may not want to share the data in the presence of family members or colleagues). However, enforcing privacy preferences that consider users' social relations may raise privacy issues, as the results of preference compliance might reveal the presence/absence of connected people (e.g., a

family member). To avoid these possible inferences, we use Private Set Intersection Cardinality (PSI-CA [160]), which allows enforcement to verify the presence of the individuals specified in the constraint without jeopardising their privacy. ConPrEF has been designed specifically for edge computing [9], an emerging and alternative paradigm to cloud computing, in which data is processed close to the device - i.e., at the network edge - without sending it remotely. Edge nodes process only a smaller amount of data from connected devices, enabling faster computation and making edge computing suitable for Industry 4.0 and real-time applications. The improvements this technology introduces, and its applicability in various sectors predict a market growth of 36.9% between 2024-2030 [161]. This expansion demands ad-hoc security and privacy-preserving mechanisms to adhere to laws and regulations (e.g., GDPR), as edge nodes typically consume user data. In this view, users should have more control over which data is collected and how it is used in edge computing by specifying privacy preferences. However, this paradigm requires further consideration when enforcing user preferences. The typical edge computing application scenario is dynamic, with users in constant motion, changing their location, the time at which they connect to the edge node, as well as the circumstances under which they connect to the node (e.g., alone, during a workout, or in an emergency situation). All these aspects might impact the user's preferences and need to be considered in such a dynamic environment [162].

Edge computing is an enabling technology for HDTs. While HDT heavily relies on continuous data acquisition from different sources, Edge Computing can act as a data collector thanks to its ability to process data closer to the source. If, on the one hand, integrating ConPrEF in edge computing allows for fast data collection and processing, on the other, it ensures that this data is handled according to the users' privacy preferences.

This chapter begins by providing a background of the employed technologies (Section 4.1). Then, it defines the concept of context-based privacy preferences and how these are modelled (Section 4.2), followed by an overview of the ConPrEF architecture (Section 4.3) and an explanation of functioning (Section 4.4). Finally, since evaluating social interaction is the most demanding operation, we have implemented a prototype of ConPrEF (Section 4.5). At first, we tested it in a realistic situation by exploiting a real dataset containing the position of edge nodes in Melbourne. Then, we did stress tests in a synthetic environment to assess our system's feasibility. As a result, we demonstrated that ConPrEF can handle realistic situations and resist computationally intensive situations.

4.1 Background

This section provides a general overview of edge computing architecture and the adopted reference scenario. Moreover, we introduce the private set intersection techniques used to enforce privacy preferences in the proposed framework.

4.1.1 Edge Computing

As shown in Fig. 4.1, edge computing relies on three layers: 1) the device layer, 2) the edge layer, and 3) the cloud layer. The *device* layer includes devices that collect data from the environment, such as devices owned by individuals (e.g., wearable devices and smartphones). The *cloud layer* provides storage and computation, allowing end users to receive personalized services based on data collected by devices. Finally, the *edge layer* is composed of multiple network devices (aka edge nodes) that act as a bridge between devices and the cloud to improve the quality and speed of services offered to end-users. These nodes can also provide services on behalf of the service provider itself without needing the users to connect to the cloud. In the following, we will interchange the terms edge nodes and service providers when discussing services and users.

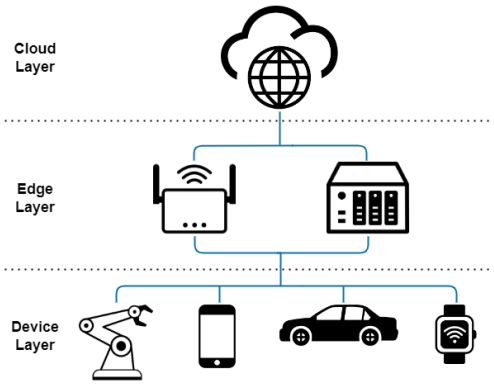


Figure 4.1: The three layers composing an edge computing network. Figure from [10].

ConPrEF reference scenario is a smart city because it is characterized by users who are often in motion and whose privacy preferences may change dynamically based on location, time, and other factors. This constant context shift may pose a challenge to the enforcement of privacy preferences. Additionally, context shifts may frequently occur in cases of rapid user movement (e.g., a user traversing the

city in a bus or by cycling). In terms of edge architecture, we model a smart city as a set of zones, each equipped with one or more edge nodes. Based on [163], we assume that each edge node covers an area of at most 1 km. At any time, users connect to the closest and most available edge node.

4.1.2 Private Set Intersection Cardinality

One of the features of ConPrEF is the possibility of preventing users' devices from sharing personal data if other specific users are in the same location, i.e., the same edge node. As it will be discussed in Section 4.2, ConPrEF enables users to specify a list of individuals that should not be on the same edge. Additionally, we assume that edge nodes maintain a list of users currently connected. The edge node and the device intersect their respective lists to determine if they share any element. To protect the privacy of users, however, the two lists and the potential users belonging to the intersection must remain confidential. For this reason, we leverage Private Set Intersection Cardinality (PSI-CA) [160] that enables the computation of the intersection of two sets, where one party (in our case, the user device) learns the cardinality of the set, whereas the other, i.e., the edge node, learns no additional information.

The literature offers different solutions for PSI-CA. ConPrEF leverages the algorithm proposed by Lv et al. [164], which exploits Bloom Filter and commutative encryption to compute the intersection cardinality. We selected this algorithm since it is fast and especially efficient when the user's data set is smaller than the edge's one.

4.2 Privacy Preferences Modelling and Storage

The primary purpose of ConPrEF is to allow users to specify their contextual privacy preferences and enforce them before sharing their data. In what follows, we first explain how we model privacy preferences and then describe a data structure we designed to allow their fast retrieval.

4.2.1 Privacy Preferences Specification

Users associate their preferences with the data to be protected. Thus, the data representation is the first relevant component of privacy preference. Similarly to [165], we assume that preferences are applied to data types (e.g., "heartbeat")

rather than specific data instances (e.g., the field 'beat' of stream heart). In particular, we model data types with different granularities and represent them in a tree structure, where the root corresponds to "all" data and the leaves are the most specific data a user can share, such as "name", "birthday", and "heartbeat". Intermediate nodes correspond to different levels of data granularities. Exploiting this tree structure, we can model the data field of a privacy preference as a pair $dataType = \langle Al_{DT}, Exc_{DT} \rangle$. By using Al_{DT}, users can specify the set of nodes they authorise to share. In particular, each node $n \in Al_{DT}$ authorises access to it and the whole subtree rooted at n. In order to handle possible exceptions to the propagation of these authorisations, users can utilise Exc_{DT} to denote the sub-nodes in the subtrees rooted at nodes in Al_{DT} for which sharing is not permitted.

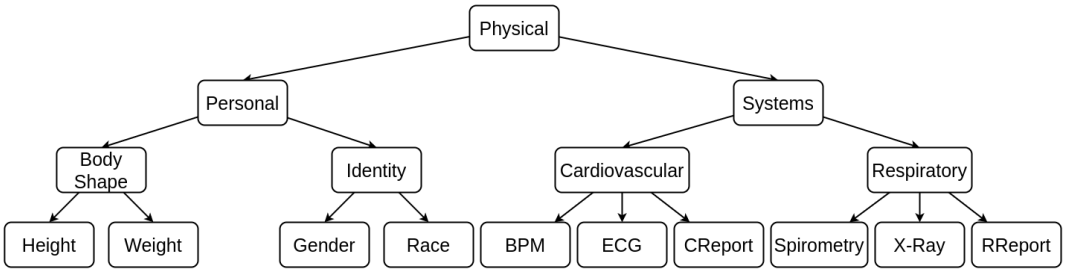


Figure 4.2: A portion of a data taxonomy. Figure from [11].

For example, let us assume a portion of a data tree as depicted in Fig. 4.2. If AL_{DT} contains "Cardiovascular", the allowed data types include all the sub-nodes: "BPM", "ECG", and "CReport". If Exc_{DT} contains "CReport", the allowed data types are only "BPM" and "ECG".

Another important component of privacy preferences is the intended purpose, which represents the reason for which users' data can be used. Similarly to other proposals, we structure purposes into a tree, adopting the structure defined in [165]. The root is the more general purpose, which includes all the others in the sub-nodes, whereas the leaves are the most specific ones. We model the intended purpose as a pair: $intPurp = \langle Al_{IP}, Exc_{IP} \rangle$, where Al_{IP} represents the allowed intended purposes, and Exc_{IP} are the exceptions. The former is the set of nodes of the purposes tree for which the user authorises the data sharing. As for the $dataType$ component, each node $p \in Al_{IP}$ authorises p and all purposes contained in the subtree rooted at p. Similarly, Exc_{IP} is the set of purposes contained in subtrees rooted at nodes in Al_{IP} for which the user does not permit data sharing.

For instance, let Fig. 4.3 be a portion of the purposes tree. Suppose Al_{IP} contains "Marketing" and the sub-nodes (all at the same level in the tree): "Offers",

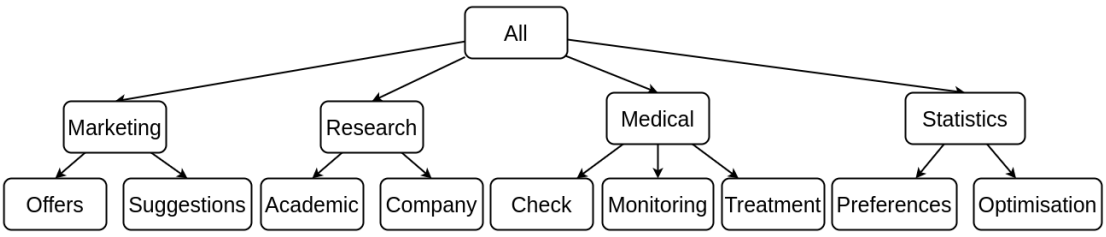


Figure 4.3: A portion of a purposes taxonomy. Figure from [11].

"Suggestions". Given Exc_{IP} equal to "Offers", $ip = \{Suggestions\}$

Formally, we define a privacy preference as follows:

Definition 1 (Privacy Preference) *A Privacy Preference is a tuple:*

$$PP = \langle dataType, allowedSrv, retPer, intPurp, thirdPart, dataBound \rangle \quad (4.1)$$

where $dataType = \langle Al_{DT}, Exc_{DT} \rangle$ *is the set of data types to which the preference applies, allowServ is the service allowed to get the data, retPer is the maximum time after which data must be removed from the system,* $intPurp = \langle Al_{IP}, Exc_{IP} \rangle$ *is the preference's intended purpose, thirdPart is the set of the third parties to which the service provider in allowServ is authorised to send data of the specified data types, whereas dataBound is a temporal boundary to specify a time interval during which the service provider can acquire the data.*

Furthermore, users' privacy preferences depend on contextual information that, in general, can be modelled by different attributes. ConPrEF considers four main contextual features, namely location, time, activities, and social relations, even if they can be easily extended to support additional ones.

The location specifies where the system can acquire users' data. We model this field as a tuple $loc = \langle city, area, \{pi\} \rangle$, where *city* is the name of the city, *area* is the coverage area of the edge node, and $\{pi\}$ is the list of points of interest in such area. Considering the smart city scenario, the defined city areas contain a set of points of interest - like restaurants, hotels, and museums.

Then, we model the time as a pair $time = \langle Al_T, Exc_T \rangle$, where Al_T is the allowed time - i.e., the time during which the system can acquire the data -, and Exc_T is the excluded time - i.e., the moment when the system cannot acquire the data within Al_T. Al_T and Exc_T can be time intervals of different granularities (e.g., hours, days, months) or a moment of the day. For instance, if a user allows data acquisition all Saturdays but in the morning, the time component of the context is modelled as $\langle Saturday, Morning \rangle$.

Furthermore, the users can specify the activities during which they allow sharing their data with the service providers. We model activities as a tree to be able to specify them with different granularities. Similarly to other fields, the activity field consists of a pair of allowed and excluded activities: $act = \langle Al_A, Exc_A \rangle$. Users can specify in Al_A the list of all activities during which they agree to share their data. On the contrary, Exc_A are the sub-nodes of nodes in Al_A containing the activities excluded from data sharing. For instance, considering Fig. 4.4, if a user sets $Al_A = sport$ and $Exc_A = indoor$, then the device can only share the user data for the activities below the node "outdoor".

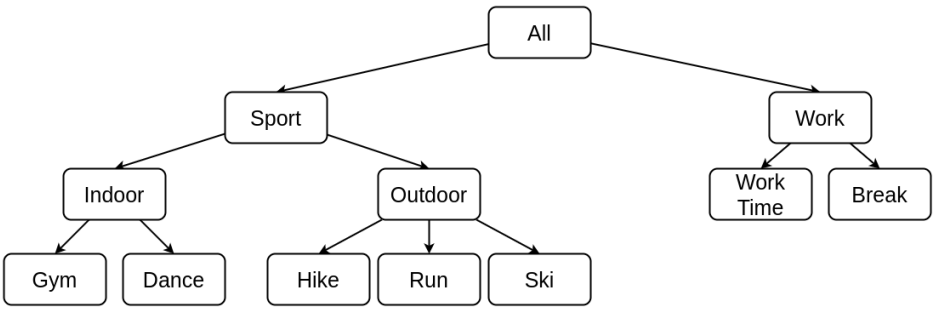


Figure 4.4: An example of activity tree. Figure from [10].

Also, users can define a list of people who prevent their devices from sharing data when connected to the same edge node. For example, a user U may wish that edges do not collect his/her data if users A and B are connected to the same edge. In addition, U may restrict data sharing if both or only one user (A or B) is present. Therefore, we model this social interaction constraint as: $blockUsers = \langle blocklist, block_{th} \rangle$, where $blocklist$ is a set of users' IDs and $block_{th}$ is an integer value, n, between 1 and the cardinality of $blocklist$, i.e., $1 \leq n < |blocklist|$. The device can share the user's data only if connected to the edge there are at maximum $block_{th}$ users in $blocklist$.

The last field in the context enables users to specify in which circumstances the regular privacy preferences can be overwritten. We name this component *sit* and assume it contains a list of situations. For simplicity, we consider the most frequent situations, i.e., "common", "emergency", and "do-not-disturb".

Considering the elements mentioned above, we formally define the context as follows:

Definition 2 (Context) *A Context is a tuple:*

$$CTX = \langle sit, loc, time, act, blockUsers \rangle$$

Where $sit = \{common, emergency, do\text{-}not\text{-}disturb\}$ *is the situation,* $loc = \langle city, area, \{pi\}\rangle$ *is the location,* $time = \langle Al_T, Exc_T\rangle$ *is the time,* $act = \langle\{Al_A\}, \{Exc_A\}\rangle$ *is the activities list, and* $blockUsers = \langle blocklist, block_{th}\rangle$ *is the social relations constraint.*

Therefore, we model a Contextual Preference as follows:

Definition 3 (Contextual Preference) *A Contextual Preference* CP *is a pair* $\{\langle C, PP\rangle\}$*, where* CTX *is a context, and* PP *is privacy preference.*

ConPrEF assumes the most restrictive preference by default in the case of preferences overlapping - for instance, when the user specifies more than one PP for a single C.

Example 1 *Consider a person, say Alice, exploiting a training application to track her workouts and suppose that she would like to authorise the sharing of training data - e.g., blood pressure (BP) and breaths per minute (BPM) - for a retention period of one year. The service provider can communicate her BP, and BPM for analysis purposes to third parties providing health services. In addition, the provider can disclose the type of workout with general vendors for personalised advertisements. The data boundary is one hour and a half, representing her typical training session length. The context where this privacy preference must be enforced is a common situation where Alice trains at the park during the weekend but not in the evening. Moreover, the park has a bar where Alice does not want to be tracked. Finally, she wants to avoid sharing data if at least one family member, among A, B and C, is with her.*

These requirements can be modelled by means of two privacy preferences, i.e., one for training data and one for workout data, associated with the same context. This forms the following contextual privacy preferences:

$$
\begin{aligned}
PP_1 &= \langle\{BP, BPM\}, training_{APP}, 1Y, \langle analysis, \emptyset\rangle, \{health_{TP}\}, 1.5H\rangle \\
PP_2 &= \langle\{workout\}, training_{APP}, 1Y, \langle adv, \emptyset\rangle, \{vendors_{TP}\}, 1.5H\rangle \\
CTX &= \langle\langle weekend, evening\rangle, \langle park, bar\rangle, \langle outdoor, \emptyset\rangle, \langle\{A, B, C\}, 1\rangle, common\rangle \\
CPs &= \{\langle PP_1, CTX\rangle, \langle PP_2, CTX\rangle\}
\end{aligned}
$$

4.2.2 CPs-tree

We assume the user's device stores the CPs in a hierarchical data structure that ConPrEF uses to facilitate the retrieval and enforcement of the subset of relevant

preferences. The data structure, named *CPs-tree*, models all CPs in a tree, where each path represents a contextual privacy preference. Given a CP, nodes in the corresponding path represent elements of its CTX and PP according to the following order: 1) $CTX.sit$, 2) $CTX.loc$, 3) $CTX.time$, 4) $CTX.act$, 5) PP, and 6) $CTX.blockUsers$.[1] This field order speeds up the retrieval and enforcement of PP, as ConPrEF has to traverse *CPs-tree* to compare the current context $\overline{CTX}$ fields against those in the tree. To this aim, we first place the fields that generate fewer sub-paths in the tree to be evaluated. In this sense, the situation is the most selective field, as it allows the device to send all user's data (emergency) or nothing (do-not-disturb) by significantly reducing the possible tree paths to cross for preference enforcement. Similarly, location and time are further selective fields.

The choice to insert the PPs before the $blockUsers$ is to prevent ConPrEF from computing the latter for services that are not required, as executing the PSI-CA is the most demanding task in terms of computation resources and time. Therefore, limiting its computation only when required increases the performance.

4.3 System Overview

We assume users sign up to the service providers of their interest and then specify their CPs through ConPrEF (cfr. Section 4.2). Those preferences are stored device-side so that the device can easily access them when needed.

ConPrEF architecture (see Fig. 4.5) includes a connection between the edge node and the service providers to indicate that the former is responsible for supplying services and identifying the users. Indeed, once a user's device interacts with a new edge node - step (1) -, the latter starts an identification phase by accessing the service provider database - step (2). Then, the edge node sends the user the list of services it offers and the related privacy policies - step (3). Finally, the device analyses the current user context by acquiring details related to location, time, activity, situation, and blocked users (cfr. Section 4.4), and uses that information to extract the PPs to enforce. The enforcement process involves the evaluation of the services' policies. If they comply with the users' PPs, the device can share the data with the edge, and the user can benefit from the service - step (4).

The device monitors the context changes by detecting, for instance, a user move or a change in the activity he/she is doing through a wearable or a fitness app (see Section 4.4 for more details). In case of any context variation or new service

[1] We use dot notation to denote selected components within a tuple.

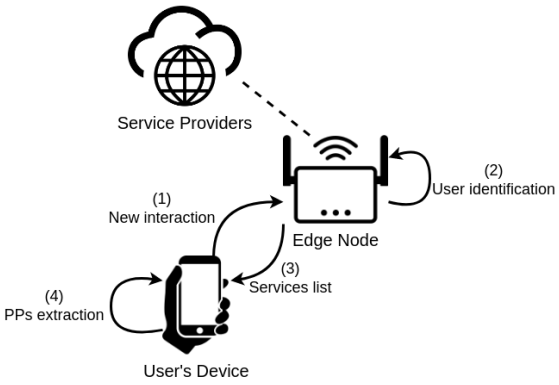


Figure 4.5: General overview of the system. Figure from [10].

requests, the device must extract again the PPs to enforce and, thus, repeat step (4).

4.4 Contextual Privacy Preferences Enforcement

Any user's device runs a process to monitor the events that trigger the enforcement of the PPs. The device executes the enforcement when the user activates a new service or when the context changes. In the first case, the device must check if the edge provides the required service. In the other case, the device is responsible for detecting context changes through periodical checks, except for $blockUsers$, as it requires the participation of the edge.

For the *time* component, we suppose the device considers the preferences' starting and ending times. We assume the device checks the location periodically, as the *loc* changes when the user changes city, area, or goes to a different point of interest. Regarding the *act* component, devices can detect a change in many different ways (e.g., a wearable device can automatically detect the beginning of a workout), or users manually activate their activities (e.g., in a training application). Similarly, for *sit*, we assume a manual setting by the users, even if we plan to complement our system with automatic detection tools in the future. Finally, for the $blockUsers$ component, assuming that the edge alerts the device each time a new user joins/exits is extremely demanding from a computational point of view. Thus, we assume the check is executed periodically (e.g., every 15 minutes) or according to the user settings.

Each device runs Algorithm 1 in the background, which waits for an event (*event*) and notifies once a change happens (Lines 1 to 3). If a new service

Algorithm 1 Monitoring

Input: An *event* triggering the re-evaluation of PPs;
$currentPPs$, the set of privacy preferences currently enforced;
$currentCtx$, the current user context;
services, the set of services that the user currently runs;

```
1: while true do
2:     wait event
3:     switch event do
4:         case newService
5:             Let newSrv be the new service
6:             currentPPs = enforcePPs(currentCtx, currentPPs, {newSrv},
                            event)
7:         case newCtx
8:             currentPPs = enforcePPs(currentCtx, currentPPs,
                            services, event)
```

($newSrv$) request is the event cause, the device executes Function *enforcePPs()* (Lines 4 to 6) passing the new service as a parameter. Otherwise, if the context changes, the device runs Function *enforcePPs()* (Lines 7 and 8) passing as a parameter the set of services that the user currently employs.

Function *enforcePPs()* (Function 1) starts setting $newPPs$ to the empty set, which will subsequently store the PPs to enforce (Line 1). Then, it stores into *CPSubTrees* all the sub-trees of the *CPs-tree* containing a privacy preference as root and whose parent nodes match the current context fields (Line 2). Subsequently, the function extracts the privacy preference *pp* from each of these subtrees ($subTree$) (Lines 3 and 4). For each service srv in the input set of services, that corresponds to the one specified in *pp*, Function *enforcePPs* extracts the $blockUsers$ component of the subtree and stores it into variable *blocked* (Lines 5 to 7). If the *blocklist* component is not empty, the device interacts with the edge (that maintains the set of connected users $conUsrs$) to compute PSI-CA and stores the intersection cardinality in *int* (Lines 8 and 9). Otherwise, or if *int* is smaller than the threshold ($block_{th}$), *pp* is added to $newPPs$ (Lines 10 and 11). Then, the function stores into $toEnforce$ the *pp*s in $newPPs$ that still need to be enforced - i.e., they are not in $currentPPs$ (Line 12). Consequently, it iteratively considers each preference *ppe* in $toEnforce$ and gathers the policy of the service contained into the privacy preference (Lines 13 and 14). If the privacy policy complies with

Function 1 $enforcePPs(ctx, currentPPs, services, event)$

Input: ctx the current user context;
$currentPPs$, the set of privacy preferences currently enforced;
$services$, the services the user wants to use;
$event$, the event triggering the re-evaluation of PPs;

1: Let $newPPs$ initialised to be empty
2: Let $CPSubTrees$ be the set of sub-trees of the device CPs-tree containing a privacy preference pp as root, obtained by traversing the paths matching ctx
3: **for all** $subTree \in CPSubTrees$ **do**
4: $pp = subTree.getPPValue()$
5: **for all** $srv \in services$ **do**
6: **if** $pp.allowedSrv == srv$ **then**
7: $blocked = subTree.getBlockUsersValue()$;
8: **if** $blocked.blocklist \neq \emptyset$ **then**
9: $int = PSICA(blocked.blocklist \cap conUsrs)$
10: **if** $int < blocked.block_{th} blocked.blocklist == \emptyset$ **then**
11: $newPPs = newPPs \cup \{pp\}$
12: $toEnforce = newPPs \backslash (newPPs \cap currentPPs)$
13: **for all** $ppe \in toEnforce$ **do**
14: Let $policy$ be the privacy policy of $ppe.allowedSrv$
15: **if** $complies(policy, ppe)$ **then**
16: $d = ppe.dataType()$;
17: $startSending([d]_{SK}, [SK]_{SPPK})$
18: $currentPPs = currentPPs \cup \{ppe\}$;
19: $toStop = currentPPs \backslash (currentPPs \cap newPPs)$
20: **for all** $pps \in toStop$ **do**
21: $d = ppe.dataType()$
22: $s = pps.allowedSrv$;
23: $stopSending(d, s)$;
24: $currentPPs = currentPPs \backslash \{pps\}$
25: **return** $currentPPs$

ppe, the function selects the type of data to be shared, executes the *startSending* function to start sending the data, and updates $currentPPs$ with ppe (Lines 15 to 18). Before being sent, the data are encrypted with a session key (SK). The session key is, in turn, encrypted with the service provider's public key ($SPPK$). We suppose the existence of a secure communication channel for data sharing

between the edge nodes and the devices willing to exploit a service.

Subsequently, Function *enforcePPs()* removes the PPs that are no longer satisfied. It collects them into *toStop* and, for each preference *pps* in *toStop*, the function extracts the user's data (d) and the service (s) specified in *pps* (Lines 19 to 22). Then, it stops sending the data to s through *stopSending*() and removes *pps* from $currentPPs$ (Lines 23 and 24). Finally, the function returns $currentPPs$ (Line 25).

Example 2 *Consider the contextual privacy preferences specified by Alice for the* $training_{APP}$ *in Example 1. Suppose she starts the application service before starting training, and her device detects her location in the park and acquires the time and situation. This event triggers Algorithm 1, whereas Alice's context is as follows:* $ctx = \{common, park, Saturday\_10{:}30a.m., running\}$. *With those data, the device executes Function enforcedPPs() giving* $training_{APP}$ *as input. Alice's CPs-tree contains the two elements:* $\langle\langle PP_1, \langle\{A, B, C\}, 1\rangle\rangle, \langle PP_2, \langle\{A, B, C\}, 1\rangle\rangle\rangle$. *Suppose that users A, B, and C are not connected to the edge node Alice is using. The device checks if the training application's policy complies with the user* PP*s to allow the data sharing and decides which data to send.*

4.5 Performance evaluation

We test ConPrEF in terms of time required by a device to perform CPs enforcement, aiming to demonstrate the feasibility of our solution in a realistic environment as well as under stress. To this aim, we execute extensive testing on two different types of devices with different computational capabilities, namely a smartphone and a wearable device (e.g., a smartwatch). Our testing environment consists of a desktop and a Raspberry Pi 3 Model B+.[2] The former has a Linux-based operating system, 16GB RAM, an Intel 11th gen. processor and 2.7TB of memory. Its use is twofold: on the one hand, it serves as an edge node; on the other, it runs a Docker container with 8GB RAM to simulate a smartphone with average technical characteristics. Instead, the Raspberry Pi mimics a wearable device, like a smartwatch.

The prototype, which is available on GitHub[3], has been developed in Python3, whereas we exploit an existing Java implementation of the PSI-CA algorithm -

[2]https://www.raspberrypi.com/products/raspberry-pi-3-model-b-plus/
[3]https://github.com/GiorgiaS/conpref

after modification to adapt it to our system - available online.[4]

For testing purposes, we consider a situation where the user moves at different speeds, and this triggers privacy preferences enforcement.[5]

Our testing includes the following phases: 1) the connection with the edge node; 2) the receipt of the privacy policies from the edge node; 3) the extraction of the user's privacy preferences correspondent to the context; and 4) the PSI-CA computation.

We test our system with both realistic and synthetic datasets. To simulate a realistic use case, we leverage the EUA dataset,[6], which contains the latitude and the longitude of many edge devices in Australia. To simplify the selection of the edge node nearest to the device, i.e., the edge to which the device is connected, and avoid overlapping, we extract from the dataset only the Telcos edge nodes.

For realistic testing, we select two paths in two distinct areas in Melbourne: the *University* - a path of $900m$ length, from Ormond College to the School of Chemistry Department - and the *Central Business District* (*CBD*) - a path of $550m$ length, from Melbourne Central Station/State Library of Victoria to Tipo 00 Restaurant. The areas involved in these paths are characterised by different edge densities (i.e., the number of edge nodes per km^2). The University path is in the area with lower density (about 36 edges per km^2), while in the CBD path, the density is higher (about 324 edges per km^2). Each user interacts with an average of 8 edge nodes in the University path and 18 in the CBD path, with an average of 82 and 96 connected users per edge node, respectively. These values derive from the population density of the two areas - about 2,948 people per km^2 in the University path,[7] and 31,000 people per km^2 in the CBD path[8] - divided by the edge density. Information about the two paths is summarised in Table 4.1.

In contrast, synthetic testing stresses our system by increasing the number of edge nodes with which the user's device must interact in a $1km$ path. Since a user interacts with 18 edge nodes in the CBD path, we start with more than double the number of edge nodes in one km, and then we increase it - i.e., 50, 75 and 100 nodes.

We extensively test our system by setting different parameters for both realistic

[4]https://github.com/liugezi/PSI-CA-Framework

[5]We only consider the position variation and not the changes of other context fields because the enforcement overhead is the same.

[6]https://github.com/swinedge/eua-dataset

[7]`https://www.arcgis.com/apps/View/index.html?appid=f937c9d1258d42f7992f748d707a88bc`

[8]https://www.abs.gov.au/statistics

Table 4.1: Users' paths in Melbourne. Table from [10].

Path	Path length	Edge density	Users\ Edge	Edge nodes
University	900m	$36/km^2$	82	8
CBD	550m	$324/km^2$	96	18

and synthetic tests. First, we consider different user speeds, simulating a user that is *walking* ($125cm/s$ [166]), travelling at a *medium* speed - e.g., a cyclist - ($20km/h$), and *fast* speed - e.g., on a scooter or by bus - ($30km/h$). The user speed affects the time available for the device to execute the CPs evaluation: the faster the user is, the less available time. We also vary the number of applications a device is simultaneously running, considering that users use nine applications per day on average.[9]. Therefore, we suppose that smartphone users interact with five services at a time and smartwatch users with at most two. We assume that each service is covered by a privacy preference. Then, we vary the number of users connected to a service. We perform realistic testing with 100 users per service, as CBD is the worst realistic situation, with 96 users per edge node. Instead, in synthetic testing, we stress our solution considering 100, 250 and 500 connected users. Finally, we consider a *blocklist* consisting of 10, 20 and 30 elements, respectively.

Table 4.2: Realistic testing results with 10 elements in the *blocklist* and 100 connected users. Table from [10].

Device	Path	Services	Time
Smartwatch	University	1	0.61sec
		2	1.00sec
	CBD	1	0.71sec
		2	1.15sec
Smartphone	University	1	0.24sec
		5	0.84sec
	CBD	1	0.23sec
		5	0.83sec

Table 4.2 shows the results of the realistic tests. The column "Time" reports the overhead introduced by ConPrEF by varying the type of device, the path and the number of services the user is exploiting simultaneously. Table 4.2 shows a

[9]https://mindsea.com/app-stats/

difference in the computation time for the two paths, which depends on the fact that in the area of the University path, having a context change is less frequent than in the CBD. To evaluate the feasibility of our approach, given the results of the tests, we evaluate the maximum time a user takes to connect from one edge to another - i.e., the maximum available time to do preference enforcement before a context change. In the worst case, i.e., when the user moves fast ($30km/h$), the available time is $13.51sec$ for the University path and $3.67sec$ for the CBD path. Therefore, the results of the tests are successful for both device types, as the worst case is the one of the smartwatches in the CBD path, which takes $1.13sec$, with an available time of $3.67sec$.

Table 4.3: Synthetic testing results with 30 elements in the *blocklist* and 500 connected users. Table from [10].

Device	Edge nodes	Services	Time
Smartwatch	50	1	0.96sec
		2	1.59sec
	75	1	1.00sec
		2	1.68sec
	100	1	1.01sec
		2	1.71sec
Smartphone	50	1	0.27sec
		5	1.00sec
	75	1	0.28sec
		5	1.05sec
	100	1	0.28sec
		5	1.06sec

Table 4.3 presents the results of synthetic tests, showing the overhead in different test settings. For space reasons, we only consider the most demanding: 500 connected users and 30 elements in the *blocklist*. Table 4.4 illustrates the time available for preference enforcement based on the number of edge nodes and the user speed. It also illustrates the maximum number of users in the path when there are 500 connected users for each edge node. By considering the two tables, we can conclude that, in our experiments, the smartphone supports ConPrEF in any situation. The smartwatch works appropriately in almost all situations, except when it executes two services simultaneously, there are more than 75 edge nodes in the path, and the user moves fast. Another aspect to consider is the maximum

Table 4.4: Available time and the total number of users based on the edge nodes in a $1km$ path and the users' speed. Table from [10].

Edge nodes	User speed	Available time	Number of users
50	Walking	16sec	25,000
	Medium	3.6sec	
	Fast	2.4sec	
75	Walking	10.67sec	37,500
	Medium	2.4sec	
	Fast	1.6sec	
100	Walking	8sec	50,000
	Medium	1.8sec	
	Fast	1.2sec	

number of users (cfr. Table 4.4), which heavily affects the PSI-CA evaluation. For instance, with 75 edge nodes, this number equals $37,500$, and having all those users in a $1km$ path is improbable.

Thanks to the exhaustive tests we performed and the obtained results, we demonstrate that ConPrEF can be applied in a realistic situation and can also successfully operate on many edge cases.

5

Efficient Privacy Compliance for HDTs

To deal with the challenge of efficient HDT views pre-materialisation, we propose a strategy, called *HDT-ViewMat*, to determine, given a workflow, which views have to be pre-materialised. This solution balances the potential delays resulting from the absence of pre-materialisations and the waste of resources after useless computations. Therefore, HDT-ViewMat assesses the user's chance of executing a task in the workflow, named *Execution Chance* (*EC*): the higher the value is, the more likely the task will be executed. The EC of a task is defined by considering the probability of the task's invocation based on the position within the workflow and the probability of the user accepting the policies of the corresponding service provider.

This first strategy is limited as it only takes into account the case where a single user, aka a single HDT, executes the process/workflow. However, if we consider metaverse applications, it is important to account for multiple users interacting with the system simultaneously (e.g., jointly attending an event or meeting). Additionally, the applications might rely on an even more complex workflow in which various providers collaborate to complete a single task. This requires changing the definition of *Execution Chance* set in HDT-ViewMat in order to incorporate multiple privacy policies linked to a task when estimating the likelihood of the user accepting all policies of the relevant service providers. A naive solution is to

compute the probability by considering among all policies associated with a task the more relaxed one, that is, the one with weaker constraints in terms of personal data usage (e.g., longer retention period, multiple usage purposes). However, this might lead to poor probability estimation and a limited number of pre-materialised views. To cope with this issue, we propose a strategy to identify a single policy, representative for the task, by solving an optimisation problem. Indeed, the goal is to select an optimal representative policy, that is, a policy that, on the one hand, minimises the costs for users, and, on the other hand, maximises the gains for service providers. We say that a user has a cost if the representative policy is more demanding than his/her privacy preference. As an example, if the policy requires authorisation for more purposes than those authorised in the user's preferences. Similarly, we say the representative policy gives a gain to a provider if it allows more than what is required by their initial privacy policy.

Finally, we were interested in knowing how people would react in the presence of the representative policy. Namely, would they be more prone to update their preferences to use a service if the policy is closer to their conditions? To achieve this aim, we deployed a survey where we asked a group of IT experts to empathise with an employee of an organisation who had to join a virtual working room to meet colleagues and clients. They had to select their preferences in two scenarios, and after observing the representative policies, they were asked to provide their opinion.

This chapter begins with Section 5.1 illustrating the proposed architecture. Then, Section 5.2 defines the concepts of Execution Chance for the single-user/provider scenario, and 5.3 extends these definitions to the multi-user/provider scope. Section 5.4 describes the functionalities of HDT-ViewMat in the two scenarios, and, Section 5.5 shows the experiments we run to test the proposed solutions. In the end, Section 5.6 describes the survey and the answers from participants.

5.1 HDT-ViewMat Overview

We consider two distinct scenarios where users undergo processes that require consuming portions of their HDT data to provide personalised services. Also, we assume the process is modelled through a workflow consisting of different tasks, where each task corresponds to a specific decision-making step executed based on the available data. The first scenario is when a single user executes the workflow, and every task is performed by a single provider. HDT-ViewMat has been conceived for this particular scenario and, in what follows, is referred to as the single-user/provider scenario. The second scenario occurs when a group of users

engage in a process where tasks may be provided by several service providers. We refer to it as a multi-user/provider scenario. In both cases, to be compliant with privacy laws, we assume that users define their preferences on personal data management via *privacy preferences*, as well as that the service providers encode their data management procedures via *privacy policies* (see Section 5.2.2 for their definitions). We further suppose that the workflow is encoded via standard process language (e.g., BPEL[1]) to be executed by a workflow engine.

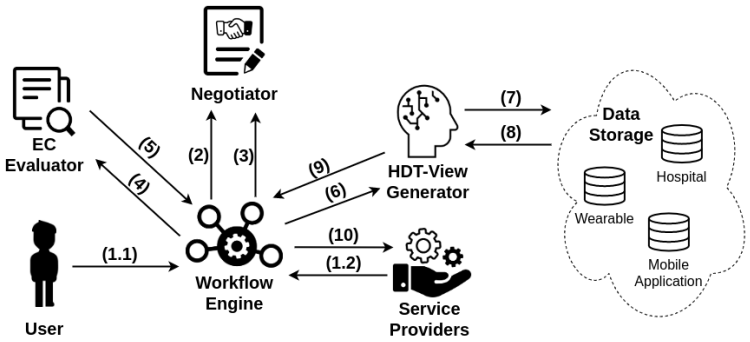


Figure 5.1: Overall architecture

As depicted in Fig. 5.1, this engine receives a request for workflow execution from the user, complemented by the privacy preferences. Before the engine initiates the execution of the workflow, the proposed architecture pre-materialises views of HDT of the requester user. In particular, the multi-user/provider scenario first requires the computation of the representative policy of each task. As described in Section 5.3.2, this is achieved by means of an optimisation algorithm executed by the negotiator component (see Fig. 5.1, steps 2-3).

Then, the engine makes decisions on which views have to be pre-materialised based on the *Execution Chance* (*EC*) measure to estimate the likelihood that a task will be executed. As described in Section 5.2.2, this estimation takes into account both the position of the task inside the workflow as well as the likelihood that the provider's privacy policy is compliant with the user's preferences. The computation of EC values is delegated to the EC evaluator component (see Fig. 5.1, steps 5-6). Based on the returned values, the workflow engine decides which views have to be pre-materialised, that is, those associated with tasks whose EC is greater than a given threshold. A reference to these tasks is then passed to the HDT-View generator (cfr. Fig. 5.1 steps 7-10).

[1]`https://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf`

During the workflow execution, the engine requires updates to the EC measures to dynamically decide which views further have to be pre-materialised based on their updated probabilities of being used. This thesis will focus on the workflow engine; before that, we introduce the Execution Chance measure as a novel metric for the selection of views to pre-materialise. In particular, we first briefly recall the EC measure defined in HDT-ViewMat for the single-user/provider scenario. Then, we focus on the multi-user/provider scenario, discussing the negotiation procedure for the selection of the task representative policy.

5.2 Execution Chance Assessment in Single-User/Provider Scenario

This section illustrates the Execution Chance in the single-user/provider scenario. This metric takes into account two factors: the user's likelihood of accepting the unique privacy policy associated with the task, and the task's position in the workflow (cfr. Section 5.2.2).

5.2.1 Reference Scenario

To support the explanation of HDT-ViewMat, in the following, we refer to the sanitary protocol defined by the European Union Standards for Tuberculosis Care (ESTC)[2] as a running example.

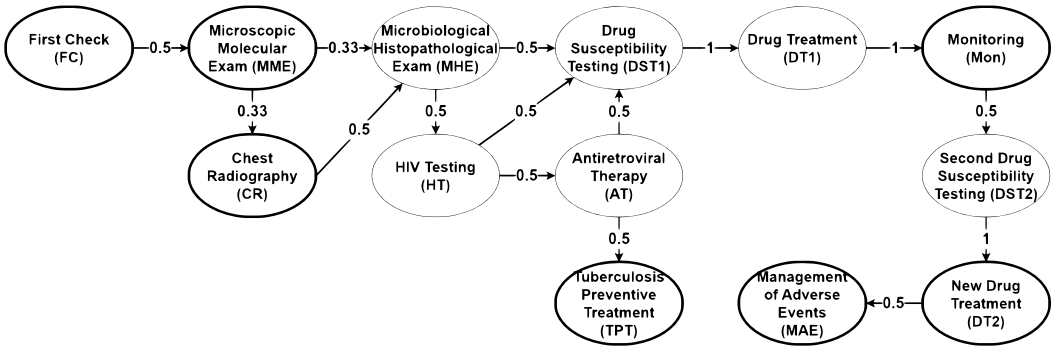


Figure 5.2: Tuberculosis process. Figure from [11].

[2] https://www.ecdc.europa.eu/en/all-topics-z/tuberculosis/prevention-and-control/european-union-standards-tuberculosis-care

Fig. 5.2 shows the graphical representation of this case study, where nodes represent tasks, and edges represent the passages between tasks. For instance, the first task is *First Check* (*FC*) followed by an initial *Microscopic and Molecular Examination* (*MME*) to identify tuberculosis and drug resistance. Nodes with a bold border are final - i.e., tasks where the protocol execution could end. We do not consider loops, assuming that if users agree to share their data with a provider for a task once, they also agree to share them at the next execution. We further assume that workflow is complemented by the probabilities of passing from one task to another, which are represented as labels on edges. In realistic scenarios, probabilities can be obtained based on heuristics (e.g., symptom history, examination results, and therapy histories of previous patients).

5.2.2 Execution Chance Assessment

To effectively pre-materialise views, HDT-ViewMat utilises the Execution Chance metric, which provides an estimation of the likelihood of executing a task inside the workflow. This measure is specifically developed to consider the likelihood that the user making the request accepts the privacy policy connected with the task and the task's position in the workflow (see Section 5.2.2).

Privacy Level Agreement

According to current laws and norms, users have to be notified and provide informed consent before the collection and use of their personal data. This consent can be automatically checked if the preferences of the user regarding his/her personal data management, as well as the privacy policy of a provider, are encoded in machine-readable format. That is, if users state a set of conditions under which the usage of their data is allowed (e.g., the purpose of data collection, third-party sharing), it is possible to automatically verify if the provider's privacy policy is compliant with them. Given a task, determining if the associated policy is compliant with the user's preference is relevant for view materialisation, aka EC estimation, as we can assume that the user will consent to the task execution (e.g., the data collection for task execution) if the policy is compliant. Nevertheless, in practical situations, users may deviate slightly from their initial preferences depending on the objective of the task or workflow they are carrying out. Therefore, the presence of a non-compliant policy does not automatically indicate that the user will not consent to using his/her data. However, we can observe that the more the policy is far from satisfying the user's preferences, the smaller the

likelihood that the user will give consent to data collection. Thus, to define EC taking into account the likelihood that the user accepts the privacy policy connected with the task, we introduce a *measure of compliance* between the privacy policies associated with a task and the user's privacy preferences, called *Privacy Level Agreement* (*PLA*). Before introducing PLA, we formally define the privacy policy and preferences. We adopt a definition based on data practice [167], where a privacy policy can be defined as a tuple $pol = \langle dt, prp, tp, rt \rangle$, stating that the provider will collect the data dt, for the purposes prp, that it could share the collected data with a set of third parties, tp, and that the time the collected data will be stored at provider servers is at maximum rt days. Privacy preferences have a similar structure as they define constraints/preferences regarding providers' data collection procedures. Specifically, a preference $pp = \langle dt, prp, tp, rt \rangle$ indicates that the user allows collecting data dt, for certain purposes prp, for a limited period of time rt, and that the data could be shared with certain third parties tp. We assume that data and purpose fields are organised in taxonomies, representing their semantic relationships and hierarchies (see Fig. 4.2 and Fig. 4.3 as examples). In contrast, the third-party component is a set, and the retention component is an integer expressing the number of days.

Example 3 *Let us consider the example in Fig. 5.2. Suppose that the specialist S performing FC task requires to collect cardiovascular reports for private research purposes, which will be kept at maximum for one year, and that the collected data could be shared with external hospitals and private clinics. The privacy policy modelling this data management is defined as* $pol_S = \langle CReport, Private, \{Hospitals, PrivClinics\}, 365 \rangle$. *In contrast, suppose that a patient, say P, agrees to share the cardiovascular reports only for academic research, with a retention period of at most nine months and prefers sharing this data only with external hospitals. These requirements can be modelled as* $pp_P = \langle CReport, Academic, \{Hospitals\}, 270 \rangle$.

Given a privacy policy and preference, the PLA between them is defined based on a set of similarity functions, each one comparing the corresponding components (i.e., dt, prp, tp, and rt), as described in the following:

- dt, prp - since data and purpose components are modelled as taxonomies, we can apply the Wu and Palmer Similarity Measure [168]. This similarity measure computes the distance between two concepts (i.e., nodes) in a taxonomy based on the number of arcs separating them. More precisely, given two nodes n_1 and n_2 in a taxonomy, $WPSim(n_1, n_2) = \frac{2*ra}{nr_1+nr_2}$, where ra is

the number of nodes between the root and the closest common ancestor of n_1 and n_2; and nr_1 (nr_2 resp.) is the number of nodes between n_1 (n_2 resp.) and the root;

- *tp* - since these are defined as sets, we use the Jaccard Similarity that, given two sets tp_1 and tp_2, evaluates their similarity as $JacSim(tp_1, tp_2) = \frac{|tp_1 \cap tp_2|}{|tp_1 \cup tp_2|}$;
- *rt* - the similarity between these components is computed through the normalised Euclidean Distance. Thus, given two retention times rt_1 and rt_2, $EucSim(rt_1, rt_2) = 1 - \frac{|rt_1 - rt_2|}{max(rt_1, rt_2)}$.

We opted for these similarity functions as they are easy to implement and are commonly used. However, alternatives could be adopted without compromising the overall logic of the system. The Privacy Level Agreement between a privacy policy and a privacy preference is formally defined as follows.

Definition 4 (Privacy Level Agreement - PLA) *Given a privacy policy pol and a privacy preference pp, we define their Privacy Level Agreement as a weighted mean:*[3]

$$PLA(pol, pp) = WPSim(pol.dt, pp.dt) * \omega_{dt} + WPSim(pol.prp, pp.prp) * \omega_{prp} + JacSim(pol.tp, pp.tp) * \omega_{tp} + EucSim(pol.rt, pp.rt) * \omega_{rt}$$

The user-defined weights ω_{dt}, ω_{prp}, ω_{tp}, and ω_{rt} represent the relative importance of each privacy component in accordance with users perspectives. Weights assume a value in the $[0, 1]$ range, and their sum must equal 1.

Example 4 *Let us consider Example 3. Both policy and privacy preferences specify the same data type $pol_S.dt = pp_P.dt = CReport$ and their similarity is $WPSim(pol_S.dt, pp_P.dt) = 1$. Subsequently, having $pol_S.prp = Private$ and $pp_P.prp = Academic$, $WPSim(pol_S.prp, pp_P.prp) = \frac{2*1}{2+2} = 0.5$ (cfr. Fig. 4.3). The third-party component of the policy is $\{Hospitals, PrivClinics\}$, whereas the one of the privacy preferences is $\{Hospitals\}$. Therefore, $JacSim(pol_S.tp, pp_P.tp) = \frac{1}{2+1-1} = 0.5$. Finally, $EucSim(pol_S.rt, pp_P.rt) = 1 - \frac{95}{365} = 0.74$. Suppose that the user assigned a higher weight to the data and purpose component (e.g., 0.3) and a lower weight to the third party and retention period ones (e.g., 0.2). Therefore, $PLA(pol_S, pp_P) = 1 * 0.3 + 0.5 * 0.3 + 0.5 * 0.2 + 0.74 * 0.2 = 0.70$*

[3]We refer to single components of a privacy policy/preference via the dot notation.

PLA measures the compliance between one policy and one preference. However, in our scenario, we are interested in measuring PLA between the set of policies and preferences applied to the data included in the view to be materialised. In particular, a task t_k executed by a provider S could require a set of data types, denoted $view(t_k)$. We assume that for each $dt \in view(t_k)$, the provider S has specified a policy on how the data type will be handled. On the other hand, the user might not have defined a preference for each $dt \in view(t_k)$. In case there is no preference for a dt, for the PLA computation, we consider the preference defined for the data type most closely related to dt, that is, the preference defined by the user for a data type having the minimum Wu and Palmer distance with dt. Let us consider, as an example, a $view(t_k)$ containing the data type *Identity* in Fig. 4.2, and assume that the user has defined privacy preferences for data *Body Shape* and *Gender*. For PLA computation, we consider the *Gender* data type as its Wu and Palmer similarity (i.e., 0.8) is greater than *Body Shape*'s distance (i.e., 0.5).

Thus, given a task, we compute a set of PLA values, one for each data type $dt \in view(t_k)$. The likelihood that a user will accept the policies linked to t_k can be approximated by calculating the average of his/her PLA values, obtaining a single value, the *Overall PLA* ($OPLA$) of t_k.

Probability of Task Invocation

To estimate the probability of invoking a task, we leverage the Discrete-Time Markov Chain Probabilistic model (DTMC) [169]. A DTMC is a model that describes the behaviour of a system that transits from one state to another with a specific probability. In particular, DTMC considers a discrete set of states $S = \{s_1, ..., s_k\}$, where the probability p_{ij} of transiting from a state s_i to state s_j does not depend on the previous states in the system. With reference to our scenario, the workflow's tasks represent DTMC states, whereas their probabilities (i.e., labels on workflow edges) indicate the transition probabilities between two consequent states. For instance, considering Fig. 5.2, the transition probability from *MME* to *MHE* is $p_{MME,MHE} = 0.33$.

In general, to estimate the probability of invoking task t_k, we need to estimate the probability of multiple transitions/steps corresponding to the path to reach t_k. In DTMC, we define the probability of transitioning from state s_i to s_j in n steps as $p_{i,j}^{(n)} = Pr(X_{m+n} = s_j | X_m = s_i)$, where X_i is a random variable representing a state in S at time step i, and m represents the time step at which the transition began. To apply the n-steps transitioning probability in our scenario, we have

to consider that a task t_k could be reached through different paths. Indeed, on the basis of input data, a workflow may generate different *execution paths*, that is, different sequences of tasks that are executed sequentially during the workflow completion. Thus, given a task t_k, we denote with $Paths(t_k)$ all the execution paths of the workflow that include t_k, that is, paths that during workflow execution require to invoke t_k. Moreover, we denote with $sbPths(t_k)$ the portion of execution paths, named subpaths, starting from the initial task t_1 and ending with the task preceding t_k. As an example, considering the task *MHE* in Fig. 5.2, the $sbPths(MHE)$ contains $path1 = FC \rightarrow MME$ and $path2 = FC \rightarrow MME \rightarrow CR$.

At last, we compute the DTMC probability of transitioning between states over multiple steps by applying the Chapman-Kolmogorov equation and considering all subpaths in $sbPths(t_k)$. This equation, which is widely used in research (e.g., process management in cloud computing [170]), computes the probability as follows: $p_{i,j}^{(m+n)} = \sum_{k \in S} p_{i,k}^{(m)} * p_{k,j}^{(n)}$, where m is the number of steps already taken and n is the number of future steps from state s_i to s_j. Thus, given a task t_k, we estimate the probability of its invocation, denoted as $prob_{t_k}$, by computing the Chapman-Kolmogorov equation, summarising the probabilities of invoking t_k in all subpaths. Let us consider Fig. 5.2 again, the Chapman-Kolmogorov formula for probability of invoking MHE from FC computes $p_{FC,MHE}^{(2)} = 0.5 * 0.33 = 0.165$ and $p_{FC,MHE}^{(3)} = 0.5 * 0.33 * 0.5 = 0.0825$ for $path1$ and $path2$, respectively. Hence, $p_{MHE} = 0.2475$.

Execution Chance

The Execution Chance of a task t_k is determined by two primary factors: (1) the likelihood of the task being executed ($prob_{t_k}$, cfr. Section 5.2.2); and (2) the likelihood of the user accepting the policies of the service provider associated with the task t_k ($OPLA(t_k)$, cfr. Section 5.2.2).

Regarding the first component, we have to consider that the event "t_k execution" is possible only if the user has consented to the privacy policies of each previous task in the execution path. Therefore, in addition to the likelihood of t_k invocation, $prob_{t_k}$, we have to consider a further element (3): the probability that the user accepts policies for all preceding tasks. This can be estimated similarly to (2), that is, by employing OPLA values assigned to previous tasks. More precisely, given a subpath $sbP \in sbPths(t_k)$ connecting the initial task t_1 to t_k, the higher the OPLAs values assigned to each task in sbP are, the higher the probability that

the user will accept policies for all preceding tasks. As such, the OPLA assigned to a single subpath sbP can be computed as the average of OPLA values of each task in sbP. More in general, the OPLA value assigned to a set of subpaths $sbPths(t_k)$, denoted as $prevOPLA(t_k)$, is computed as the average of OPLA assigned to each single subpath $sbP \in sbPths(t_k)$.

$$prevOPLA(t_k) = \frac{1}{|sbPths(t_k)|} \sum_{sbP \in sbPths(t_k)} \left(\frac{1}{|sbP|} \sum_{t_x \in sbP} OPLA(t_x) \right)$$

Before presenting the Execution Chance definition, we introduce a further element that affects the second component (2), that is, the likelihood of the user accepting the policies of the service provider associated with the task t_k, $OPLA(t_k)$. Indeed, it is crucial to acknowledge that users may choose to deviate from their usual privacy settings in specific situations. In particular, this situation may arise when they are approaching the end of a process, and the violation of a privacy preference relates to the last tasks of that workflow. This could potentially have an impact on the Execution Chance of t_k. To this end, we introduce a weight α to adjust the $OPLA(t_k)$ value. In particular, if task t_k is near the end of the process, the user could be more likely to relax his/her preferences and accept the policies because he/she has already executed most of the tasks. In this case, the weight should decrease the value of $OPLA(t_k)$. As such, we defined α based on the distance of t_k to the end of an execution path in $Paths(t_k)$. In particular, given a $path$ in $Paths(t_k)$, α is defined as relative position of t_k with respect to $path$, that is, $\frac{pos(path,t_k)}{|path|+1}$, where $pos(path, t_k)$ returns the position of t_k in $path$. Considering that several execution paths, i.e., $Paths(t_k)$, could pass via t_k, we can compute distinct α value for each path in $Paths(t_k)$ and then compute their average as follows.

$$\alpha_{t_k} = \frac{1}{|Paths(t_k)|} \sum_{path \in Paths(t_k)} \frac{pos(path, t_k)}{|path| + 1}$$

We can define the concept of Execution Chance.

Definition 5 (Execution Chance - EC) *Given a task t_k in a workflow, its Execution Chance (EC) is defined as follows:*

$$EC(t_k) = prevOPLA(t_k) * \Big(OPLA(t_k) + \alpha_{t_k} * (1 - OPLA(t_k)) \Big) * prob_{t_k}$$

Example 5 *Considering the example in Fig. 5.2, let us assume to evaluate the*

EC for TPT task, supposing $OPLA = 0.9$ *for each task. According to Fig. 5.2,*

$$Path(TPT) = \{FC \rightarrow MME \rightarrow MHE \rightarrow HT \rightarrow AT,$$
$$FC \rightarrow MME \rightarrow CR \rightarrow MHE \rightarrow HT \rightarrow AT\}$$
$$\alpha_{TPT} = \frac{1}{2} * \left(\frac{6}{7} + \frac{7}{8}\right) = 0.87$$
$$prevOPLA(TPT) = \frac{1}{2}\Big[\frac{1}{5}(OPLA(FC) + OPLA(MME) + OPLA(MHE) + OPLA(HT) + OPLA(AT)) + \frac{1}{6}(OPLA(FC) + OPLA(MME) + OPLA(CR) + OPLA(MHE) + OPLA(HT) + OPLA(AT))\Big]$$
$$prob_{TPT} = (0.5 * 0.33 * 0.5 * 0.5 * 0.5) + (0.5 * 0.33 * 0.5 * 0.5 * 0.5 * 0.5) = 0.03$$
$$EC(TPT) = prevOPLA(TPT)\big[OPLA(TPT) + \alpha_{TPT}(1 - OPLA(TPT))\big]prob_{TPT}$$
$$= 0.9 * \big(0.9 + 0.87 * (1 - 0.9)\big) * 0.03 = 0.03$$

5.3 Execution Chance Assessment in Multi-User/Provider Scenario

In this section, we introduce the execution chance for the multi-user/provider scenario. As described in Section 5.1, an important difference with respect to a single-user/provider scenario is that, given a task, multiple policies and preferences need to be taken into account. We first describe the reference scenario to support the explanation of the new strategy (Section 5.3.1). Then, we introduce the strategy for selecting the representative policy for the task, which is done by the policy negotiator component (Section 5.3.2), and then we present the new definition of EC (Section 5.3.2).

5.3.1 Reference Scenario

To complement the explanation of the integrations to the EC metric, we consider the metaverse education case as a running example for the multi-user/provider scenario. We assume online courses where a group of users can attend classes, study in groups, and participate in laboratory activities. The course also offers the possibility of attending a virtual school trip, where students can apply their learning in a real situation, simulated in the metaverse.

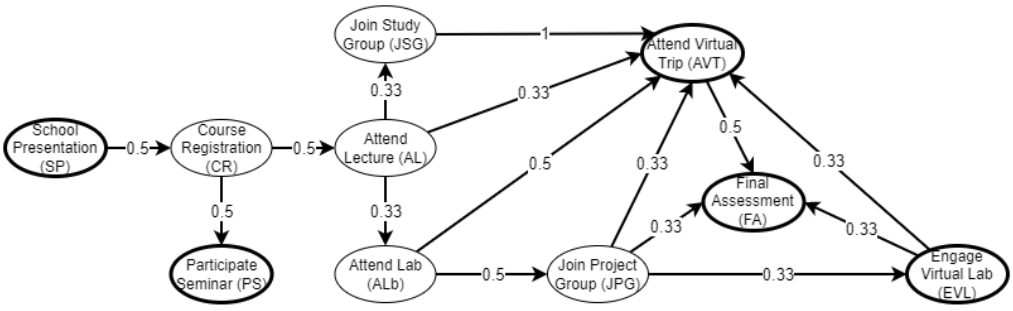


Figure 5.3: Metaverse Education Process

Fig. 5.3 shows the workflow of this case study, where nodes represent tasks, and edges represent the dependency between tasks. For instance, the first task is *School Presentation* (*SP*) followed by an initial *Course Registration* (*CR*). As for the single-user/provider scenario, we do not consider loops.

5.3.2 Execution Chance Assessment

In this section, we introduce the execution chance for the multi-user/provider scenario. As described in Section 5.1, an important difference with respect to a single-user/provider scenario is that, given a task, multiple policies and preferences need to be taken into account. We first introduce the strategy for selecting the representative policy for the task, which is done by the policy negotiator component, and then we present the new definition of EC.

Policy Negotiator

This component is designed to compute the representative policy for each task within the workflow. The goal is to select an optimal representative policy for t_k, that is, a policy that, on the one hand, minimises the costs for users, and, on the other hand, maximises the gains for service providers. To be more specific, we determine each component of the representative policy (namely, *prp*, *tp*, and *ret*) by solving a many-objective optimisation problem that involves four objective functions [171].

We start the analysis from the purpose field *prp*. Hereafter, given a task t_k requiring access to data dt, we denote as Pol^{dt} the set of policies applied on dt and defined by service providers executing t_k, and with Prp_{SP} the bag of purposes specified in Pol^{dt}. Similarly, we denote as PP^{dt} the set of privacy preferences

defined for data dt by users running t_k, and with Prp_U the bag of purposes contained in PP^{dt}. We identify four objective functions: two for the user side and two for the service provider side. The first user-side function aims to maximise user savings by ensuring that the representative policy requests fewer purposes than those that users specify. The second aims to minimise the costs for users in terms of reduced effectiveness of their privacy preferences. Before presenting these functions, we need to introduce two metrics used to measure the savings and the costs. Let's denote with $\overline{p}$ the purpose of a potential representative policy, where $\overline{p} \in Prp_U \cup Prp_{SP}$. With respect to $\overline{p}$, we define:

- User-Favourable purposes, $UF_{\overline{p}}$, as the bag of purposes, containing those purposes in Prp_U that in the purpose taxonomy are positioned in the path connecting $\overline{p}$ to the root. As an example, referring to Fig. 4.3, $UF_{Academic} = \{Research, All\}$. These are purposes that, if included in the user preference, are less restrictive than $\overline{p}$ and thus they bring savings/benefits to the users.

- User-Unfavourable purposes, $UU_{\overline{p}}$, as the bag of purposes containing those purposes in in Prp_U that in the taxonomy belongs to a subtree rooted by $\overline{p}$ or belong to different taxonomy branches. These are purpose values that, if included in the user preference, are more restrictive than $\overline{p}$. Thus, in case the intermediate policy has $\overline{p}$ as purpose, these values represent a cost for users.

- User-Equal purposes, $UE_{\overline{p}}$, as the bag of purposes containing all the elements in Prp_U that are equal $\overline{p}$.

The first user-side function, f_{prp}, serves to quantify the savings generated by $\overline{p}$, and as such, it must be optimised.

$$f_{prp}(\overline{p}, UF_{\overline{p}}, UE_{\overline{p}}) = \omega_1 \frac{|UE_{\overline{p}}|}{|Prp_U|} + \omega_2 \frac{1}{|UF_{\overline{p}}|} \sum_{p_f \in UF_{\overline{p}}} \Big(1 - WPS(\overline{p}, p_f)\Big)$$

where ω_1, ω_2 are weights defined by the negotiator, which sum equals 1.

This function calculates the average savings of all favourable purposes, $p_f \in UF_{\overline{p}}$. This is measured as the distance of p_f from $\overline{p}$ (e.g., 1 - Wu and Palmer similarity value). Indeed, considering that p_f is an ancestor of $\overline{p}$, the greater the distance between p_f and $\overline{p}$, the more purposes associated with descendants are authorised by p_f, thus the greater the user savings in case $\overline{p}$ is selected. The function also incorporates the number of equal purposes $UE_{\overline{p}}$, which do not denote savings

but also do not incur costs, so they are weighted differently in the calculation to reflect their different impacts on the overall outcome.

The second optimisation function, g_{prp}, represents the users' costs and thus needs to be minimised. It considers all unfavourable purposes in $UU_{\overline{p}}$ as follows:

$$g_{prp}(\overline{p}, UU_{\overline{p}}) = \frac{1}{|UU_{\overline{p}}|} \sum_{p_u \in UU_{\overline{p}}} \Big(1 - WPS(\overline{p}, p_u)\Big)$$

The function estimates the costs as the average distance of p_u from $\overline{p}$. As p_u is a descendant (or belonging to another branch, resp.), the greater the distance between p_u and $\overline{p}$ is, the more purposes associated with descendants of $\overline{p}$ are authorised (or the more semantically distant the authorised purposes, from another branch, are from p_u, resp.). This distance represents costs in case $\overline{p}$ is selected.

On the service-provider side, the first function aims to maximise the providers' gain by ensuring that they will be authorised for more purposes than those specified in their policies, and the second aims to minimise the loss of providers in case they are authorised for fewer purposes. Also for these objective functions, we need to introduce two metrics used to measure the gains and the losses. Let $\overline{p}$ be the purpose of a potential representative policy, we define:

- Provider-Favourable purposes, $PF_{\overline{p}}$, as the bag of purposes containing those purposes in Prp_{SP} that, in the taxonomy, are included in the subtree rooted by $\overline{p}$. Thus, if $\overline{p}$ is selected for the representative policy, providers will be authorised for more purposes than those specified in their initial policies, as $\overline{p}$ is an ancestor for them.

- Provider-Unfavourable purposes, $PU_{\overline{p}}$, as the bag of purposes containing those purposes in Prp_{SP} positioned in the path connecting $\overline{p}$ to the root or belonging to different branches in the taxonomy. Contrary to the previous case, here providers will be authorised for fewer purposes (as $\overline{p}$ is a descendent of purposes in the initial policies) or to purposes far from what they required in the initial policies.

- Provider-Equal purposes, $PE_{\overline{p}}$, the bag of purposes containing those purposes in Prp_{SP} equal to $\overline{p}$.

The first server-side objective function, h_{prp}, quantifies the gains of those providers that will be authorised for more purposes than what they specified in their policies.

$$h_{prp}(\overline{p}, PF_{\overline{p}}, PE_{\overline{p}}) = \omega_1 \frac{|PE_{\overline{p}}|}{|Prp_{SP}|} + \omega_2 \frac{1}{|PF_{\overline{p}}|} \sum_{p_f \in PF_{\overline{p}}} (1 - WPS(\overline{p}, p_f))$$

Similarly, to the first user-side function, it computes the average gains of all favourable purposes, $p_f \in PF_p$, as the distance of p_f from $\overline{p}$. The function also incorporates the number of equal purposes $PE_{\overline{p}}$.

The second server-side optimisation function represents the providers' loss, which has to be minimised. It considers all unfavourable purposes in $PU_{\overline{p}}$ as follows:

$$e_{prp}(\overline{p}, PU_{\overline{p}}) = \frac{1}{|PU_{\overline{p}}|} \sum_{p_u \in PU_{\overline{p}}} \Big(1 - WPS(\overline{p}, p_u)\Big)$$

The purpose value prp_N, for the representative policy, is then selected as the optimal solution of the optimisation problem. This is the value $\overline{p} \in Prp_U \cup Prp_{SP}$ that maximises functions f_{prp} and h_{prp} and minimises g_{prp} and e_{prp}. Nevertheless, the task of identifying the optimal solution is complex because of the difficulties in concurrently satisfying all four functions. Indeed, multi-objective optimisation algorithms return a set of optimal solutions instead of just one. Consequently, it is necessary to establish a criteria to choose the solution to be selected among the alternatives provided. To this purpose, we adopt the NSGA-III algorithm [171], which analyses purposes in $Prp_U \cup Prp_{SP}$ and returns a subset $\overline{P}$ of optimal solutions. Then, as selection criteria, we aim to choose the purpose that balances users' and providers' benefits and costs. In particular, we calculate a score for each $\overline{p} \in \overline{P}$, by computing the differences between users' benefits and costs and the difference between providers' benefits and costs. We define the score as the distance between these two differences, as follows:

$$Score_{\overline{p}} = (f_{prp} - g_{prp}) - (h_{prp} - e_{prp})$$

Purposes with a score value = 0 are those that provide a balance between users and providers. Thus, from those purposes in $\overline{P}$, we select as prp_N, the one whose score is nearest to 0.

The computation of the third-party field tp_N of the representative policy is analogous to that of prp_N, with the difference that this field is defined as a set (cfr. Section 5.2.2). Hereafter, given a task t_k that requires access to data dt, we refer to the bag of third-party sets contained into users' preferences in PP^{dt} (in policies in Pol^{dt}, resp.) as TP_U (TP_{SP}, resp.).

Similarly to prp_N, we introduce some metrics used by the objective functions. In particular, given $\overline{tp} \in TP_U \cup TP_{SP}$, representing the third-party component of a possible representative policy, we define:

- User-Favourable third-parties, $UF_{\overline{tp}}$, as the bag of third-party sets belonging to TP_U that includes $\overline{tp}$. If $\overline{tp}$ is chosen, users with third parties in $UF_{\overline{tp}}$ have benefits, as $\overline{tp}$ does not authorise additional third parties beyond those that have already been authorised by their initial preference.

- User-Equal third-parties, $UE_{\overline{tp}}$, as the bag of third-party sets contained in TP_U that are equal to $\overline{tp}$.

- User-Unfavourable third-parties, $UU_{\overline{tp}}$, as the bag of third-party sets that are not included in $UF_{\overline{tp}}$ or in $UE_{\overline{tp}}$. If $\overline{tp}$ is selected, users with third parties in $UU_{\overline{tp}}$ incur costs, as $\overline{tp}$ authorises additional third parties beyond those that have already been authorised.

- Provider-Favourable third-parties, $PF_{\overline{tp}}$, as the bag of third-party sets in TP_{SP}, which are a subset of $\overline{tp}$.

- Provider-Equal third-parties, $PE_{\overline{tp}}$, as the bag of third party sets in TP_{SP}, that are equal to $\overline{tp}$.

- Provider-Unfavourable third-parties, $PU_{\overline{tp}}$, as the bag of third-party sets in TP_{SP} that are not included in $PF_{\overline{tp}}$ or in $PE_{\overline{tp}}$.

Similarly to the purpose, we define four objective functions: two user-sides (i.e., f_{tp} and g_{tp}) and two provider-sides (i.e., h_{tp} and e_{tp}). As distance measure, we exploit the Jaccard similarity, $JS()$, as follows:

$$f_{tp}(\overline{tp}, UF_{\overline{tp}}, UE_{\overline{tp}}) = \omega_1 \frac{|UE_{\overline{tp}}|}{|TP_U|} + \omega_2 \frac{1}{|UF_{\overline{tp}}|} \sum_{t_f \in UF_{\overline{tp}}} \Big(1 - JS(\overline{tp}, t_f)\Big)$$

$$g_{tp}(\overline{tp}, UU_{\overline{tp}}) = \frac{1}{|UU_{\overline{tp}}|} \sum_{t_u \in UU_{\overline{tp}}} \Big(1 - JS(\overline{tp}, t_u)\Big)$$

$$h_{tp}(\overline{tp}, PF_{\overline{tp}}, PE_{\overline{tp}}) = \omega_1 \frac{|PE_{\overline{tp}}|}{|TP_{SP}|} + \omega_2 \frac{1}{|PF_t|} \sum_{t_f \in PF_t} \Big(1 - JS(\overline{tp}, t_f)\Big)$$

$$e_{tp}(\overline{tp}, PU_{\overline{tp}}) = \frac{1}{|PU_{\overline{tp}}|} \sum_{t_u \in PU_{\overline{tp}}} \Big(1 - JS(\overline{tp}, t_u)\Big)$$

As for prp_N, to select the third-party component tp_N for the representative policy, we run the NSGA-III algorithm [171]. This elaborates all values in $TP_U \cup TP_{SP}$ and returns the set of optimal third-party sets $\overline{TP}$. We then select the one with the score nearest to 0 (computed with a formula similar to $Score_{\overline{tp}}$).

The last field for the representative policy is the retention period. Since this component is defined as an integer, we calculate the value rt_N to be inserted into the representative policy by averaging the retention values specified in policies Pol^{dt}, stated by providers executing task t_k on data dt, and values specified in preferences PP^{dt}, stated by users running t_k. More precisely, we compute the average of retention values in Pol^{dt} and the average of retention values in PP^{dt}. rt_N is set as the average of them.

The negotiator composes the representative policy pol_N for task t_k on data dt as follows: $\langle dt, prp_N, tp_N, rt_N \rangle$. Finally, the set of representative policies for t_k, denoted as $Pols_N(t_k)$, includes the pol_N for all data types required by the service providers executing t_k.

Table 5.1: Privacy policies on heartbeats data type, Example 6

Service Provider	Purpose	Retention	Third Party
SP1	Marketing	180	Ad Network, Legal Advisor, Cloud Storage Provider, Market Research Firm, Customer Support Tool
SP2	Statistics	100	Cloud Storage Provider, Email Marketing Platform, Legal Advisor, Analytics Provider

Table 5.2: Privacy preferences on heartbeats data type, Example 6

User	Purpose	Retention	Third Party
U1-PP1	Offers	30	Legal Advisor
U2-PP1	Marketing	100	Ad Network, Legal Advisor, Cloud Storage Provider
U1-PP2	Optimisation	150	Cloud Storage Provider, Email Marketing Platform
U2-PP2	Optimisation	90	Legal Advisor, Cloud Storage Provider, Analytics Provider

Example 6 *Suppose that the workflow depicted in Fig. 5.3 is executed by two users, each one with two privacy preferences. Moreover, suppose that two service providers are participating in supplying task SP. Table 5.1 and Table 5.2 show the privacy preferences and providers' policies defined for the heartbeat. To define the representative policy for task SP, the negotiator executes the NSGA-III algorithm. As an example, the algorithm for the third-party component returns three solutions: 1) {Cloud Storage Provider, Analytics Provider, Customer Support Tool, Legal Advisor, Ad Network}, with score = 0.19; 2) {Email Marketing Platform, Market Research Firm, Analytics Provider, Customer Support Tool, Cloud Storage Provider, Legal Advisor, Ad Network}, with score = 0.08; and 3) {Cloud Storage Provider, Legal Advisor, Email Marketing Platform}, with score = 0.04. Since the latter has the score closest to 0, the negotiator selects it as the third-party field of the representative policy. Then, the final representative policy is the following:*

$$pol_N = \langle HeartBeats, Marketing, 116, \{Email\ Marketing\ Platform, Cloud\ Storage\ Provider, Legal\ Advisor\}\rangle$$

Multi-User/Provider Execution chance

The design ideas outlined in Section 5.2.2 for single-user/provider scenarios are applicable to multi-user/provider cases as well. Given a task t_k, the only differences are that: (1) we compute distinct execution chance values, EC_u, for each user u running t_k; (2) the privacy level agreement PLA (i.e., Overall Privacy Level agreement, $OPLA(t_k)_u$) is computed by considering u's privacy preference and the representative policy defined by the negotiator for t_k.

Definition 6 (Execution Chance - EC -Multi-User/Provider) *Given a task t_k in a workflow, and a user u running t_k, the Execution Chance of t_k is defined as follows:*

$$EC(t_k)_u = prevOPLA(t_k)_u\Big(OPLA(t_k)_u + \alpha_{t_k}(1 - OPLA(t_k)_u)\Big)prob_{t_k}$$

5.4 Workflow Engine

In general, a workflow engine aims at properly invoking tasks of the workflow based on the execution order specified in the workflow definition. In HDT-Views, the workflow engine has been modified to add a preliminary computation of EC values, as well as the continuous update of these values, to determine the views to be pre-materialised, that is, those associated with a task with EC greater than a threshold. The procedure executed by the workflow engine in the multi-user/provider

scenario is the same as that in the single-user/provider scenario. The only difference is the extension incorporating the policy negotiation component before the EC assessment.

Algorithm 2 EngineExecution

Input: $Pols$, privacy policies of involved providers;
PPs, privacy preferences of requesting users;
th, a threshold;
bpel, the BPEL document encoding the workflow.

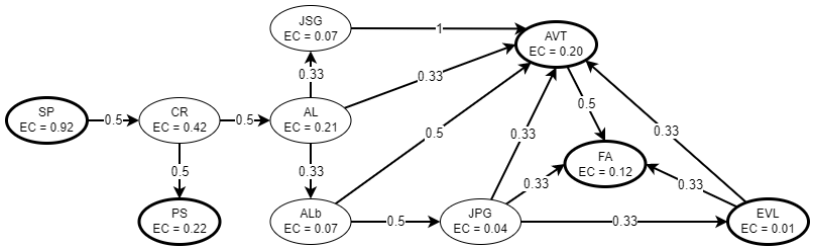
1: $Pols_N \leftarrow negotiation(Pols, PPs)$
2: $enbpel \leftarrow ECEvaluation(Pols_N, PPs, bpel, t_1)$
3: $toMaterialise = \{t \in enbpel | EC(t) \geq th\}$
4: $HDT\text{-}VGenerate(toMaterialise)$
5: $materialised = toMaterialise$
6: Wait invocation t_1
7: Invoke t_1
8: $currentTsk = t_1$
9: **while** processIsRunning **do**
10: **if** $currentTsk \neq t_1$ **then**
11: $enbpel \leftarrow ECEvaluation(Pols_N, PPs, bpel, currentTsk)$
12: $Updatematerialise = \{t \in enbpel | EC(t) \geq th\}$
13: $toMaterialise = Updatematerialise - materialised$
14: $HDT\text{-}VGenerate(toMaterialise)$
15: $materialised = materialised \bigcup toMaterialise$
16: Wait termination $currentTsk$
17: Let $newTsk$ be the task to be invoked after $currentTsk$ in *bpel*
18: Wait invocation $newTsk$
19: $currentTsk = newTsk$

Algorithm 2 shows the complete procedure performed by the engine. As for the single-user/provider scenario, the algorithm takes the BPEL representing the workflow (*bpel*) and the threshold value (th) as input. Then, for each task t in the workflow, it also receives as input the set of policies ($Pols_t$) of the service providers executing t and the users' privacy preferences (PPs_t) - $Pols_t$ contains the policies of a single service provider and PPs_t the preferences of an individual user in the single-user/provider scenario. For brevity, in the algorithm, we denote with $Pols$ and PPs the set of policies and preferences of all tasks associated with the workflow.

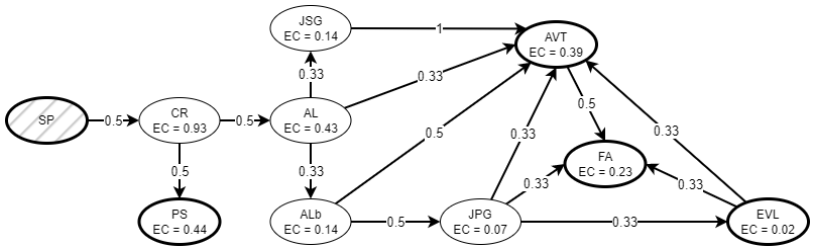
The algorithm begins by determining the tasks' representative policies through the *negotiation*() function (Line 1) - this step is skipped in the single-user/provider case. For each task, it analyses the corresponding set of policies and preferences in $Pols$ and PPs and computes their representative policies. The resulting set of policies is denoted as $Pols_N$. From this point on, Algorithm 2 behaves similarly for the two scenarios. The only difference is in the multi-user/provider, which passes $Pols_N$ to the $ECEvaluation()$ function as the set of policies used to assess the OPLA and, thus, EC values. Instead, the algorithm passes $Pols$ in the single-user/provider scenario. Therefore, $ECEvaluation()$ function akes as input $Pols_N/Pols$, Pps, *bpel*, and t_k, a reference to the current considered task in *bpel* (e.g., task ID). In line Line 2, t_k corresponds to the starting task t_1 of the workflow. The function returns an enhanced BPEL version, *enbpel*, where all tasks reachable from t_k, t_k included, are complemented with the corresponding EC and OPLA values. We define a task t_r as reachable from t_k if it belongs to at least one path in $Paths(t_k)$ and it is after t_k. Note that the execution of $ECEvaluation()$ with t_1 implies the computation of EC and OPLA values for each task in *enbpel*.

From the *enbpel* file, the algorithm extracts the set of tasks ($toMaterialise$) with an EC greater than (or equal to) threshold th (line Line 3). The views corresponding to these tasks have to be pre-materialised and thus are passed to the HDT-View generator (Line 4). The variable *materialised* stores the references to tasks whose views have been materialised (Line 5). The engine then waits for the request to invoke the first task, t_1, and invokes it as a consequence (Line 6 and Line 7). Until the engine reaches the end of the workflow, it continuously waits for the request for the invocation of a new task, which arrives upon the completion of the current task (While loop in Line 9). Once it arrives, the algorithm updates the EC and OPLA values by re-executing the $ECEvaluation()$ function with reference to the current task and passing $Pols_N$ in the multi-user/provider scenario, and $Pols$ otherwise (Line 11). It determines the set of views to be materialised, that is, those associated with tasks with EC greater (or equal) than the threshold and that have not yet been materialised (Line 12 and Line 13). These are passed to HDT-View Generator (Line 14), and the variable *materialised* is updated accordingly (Line 15).

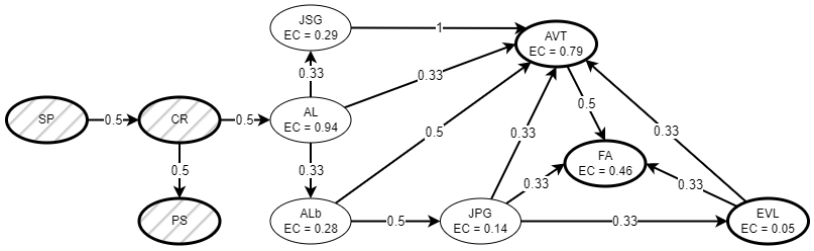
Example 7 *Consider the workflow in Fig. 5.3 - for this example, we take the multi-user/provider scenario, but we can consider the single-user/provider (Fig. 5.2) indistinguishably -, where we replaced the names of tasks with their acronyms and reported EC values inside nodes. This figure shows an example of the initial EC*



(a) Example of EC evaluation for a user



(b) Dynamic evaluation when the user is executing task CR



(c) Dynamic evaluation when the user is executing task AL

Figure 5.4: Graphical representation of EC evaluation of Metaverse Education Process example. Figure from [11].

computation for a single user, performed by ECEvaluation() (line Line 2 of Algorithm 2). Values are assessed by assuming that the OPLA of each task is set to 0.9*. Overall, we can observe that this first EC evaluation generates smaller values for tasks at the end of the path (e.g., JPG and EVL). Let us assume that users proceed in the workflow by invoking task CR with a consequent update of the EC values. Fig. 5.4b shows the ECs after their new assessment. First of all, EC values of tasks following CR are greater than those generated by the initial evaluation. It is worth noting that task AVT, from an initial* $EC(AVT) = 0.20$*, passes to* $EC(AVT) = 0.39$*. Suppose now task AL is invoked. The Algorithm 2 updates EC values (Fig. 5.4c), where values of tasks following AL further increase their corresponding values compared to the previous evaluation (For instance, the EC*

of task AVT reaches a value of 0.79. *Consider now the EC evaluation illustrated in Fig. 5.4 and a threshold* $th = 0.2$. *After executing the initial evaluation, only four tasks are pre-materialised, namely SP, CR, PS, AL, and AVT. When CR is invoked, the updated EC values also prompt the pre-materialisation of views for task FA. Later, once AL is invoked, the EC values of JSG and ALb exceed th.*

5.5 Experimental Evaluation

This section presents the experiments run to evaluate the efficiency of HDT-ViewMat in both single-user/provider and multi-user/provider scenarios. We first illustrate the results for the first scenario and then provide an overview of the outcomes in the multi-user/provider scenario.

5.5.1 Single-User/Provider Analysis

This section presents a series of tests to assess the efficacy of HDT-ViewMat applied to the single-user/provider case. The evaluation is conducted using two datasets: real-world processes generated considering the healthcare scenario and a benchmark of BPEL processes. In the following, we present results on the BPEL benchmark as long as those obtained on the healthcare scenario.

BPEL Benchmark Dataset

We considered the real-world BPEL processes benchmark from [172] by filtering out those processes that represent a unique sequence of task invocations since they are not relevant for testing the pre-materialisation strategy as all tasks will be most likely executed. The final set consists of 27 BPEL processes that we then organised in groups based on the number of execution paths and, thus, the number of possible views (aka tasks in the path) to be materialised. Features of resulting clusters are represented in Table 5.3, where for each *Cluster*, we present the number of its *#Processes*, the number of execution *#Paths* of these processes (i.e., the minimum and maximum number of paths among all processes), and the number of *#Tasks* in paths (i.e., the minimum and maximum number of tasks among all paths). For instance, cluster C7 contains three processes, each with a number of paths in the interval $5 - 6$ and tasks between $6 - 13$.

Each process in the dataset is further complemented with privacy preferences and policies to compute OPLA for each task. In doing that, we acknowledge

Table 5.3: Clusters description of the benchmark dataset. Table from [11].

Cluster	#Paths	#Tasks	#Processes
C1	2	3-4	4
C2	2	5-10	3
C3	3	3-5	3
C4	3	7	2
C5	4	2-5	4
C6	4	6-13	3
C7	5-6	6-13	3
C8	10-12	6-12	3
C9	44	110	1
C10	45	44	1

that different users might have different attitudes with respect to privacy and thus generate privacy preferences that are more or less stringent. This impacts the OPLA values. Indeed, given a task and corresponding privacy policy, a more privacy-conscious user might have preferences with more stringent constraints and, thus, an OPLA value smaller than those obtained when comparing the privacy preferences of users with fewer constraints. In particular, we used [173] to model four different types of users according to their privacy concerns: Unconcerned (*Type1*), Circumspect (*Type2*), Wary (*Type3*) and Alarmed (*Type4*). We assume that each user type corresponds to a different range of OPLA values. As described [173], users of Type1 register to any website, and, most of the time, they share accurate information (i.e., $OPLA = [0.83, 1]$); Type2 users pay more attention but are still permissive (i.e., $OPLA = [0.50, 0.82]$); Type3 users pay attention to registering on any website and avoid sharing accurate personal information (i.e., $OPLA = [0.18, 0.49]$); Type4 users register to a website only if it is strictly necessary (i.e., $OPLA = [0, 0.17]$).

For testing purposes, we simulate that each process of the benchmark is executed four times, one per user type. At each simulation, all tasks are assigned an OPLA value randomly taken from the range corresponding to the considered user type.

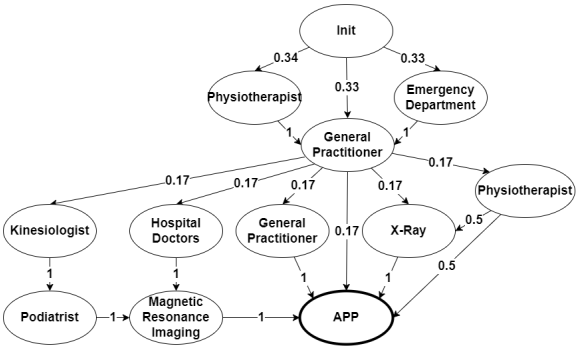
Healthcare dataset

The healthcare scenario was further considered to generate three additional BPEL processes representing real case studies.

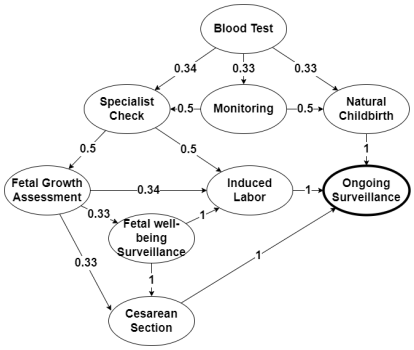
Advanced Practitioner Physiotherapist (APP), Fig. 5.5a. This process is taken from [174], a survey exploring patients' clinical journeys attending APP services in two different hospitals.

Gestational Diabetes (GD), Fig. 5.5b. It is based on the International Federation of Gynecology and Obstetrics guidelines for diagnosis, management and care of gestational diabetes mellitus [175].

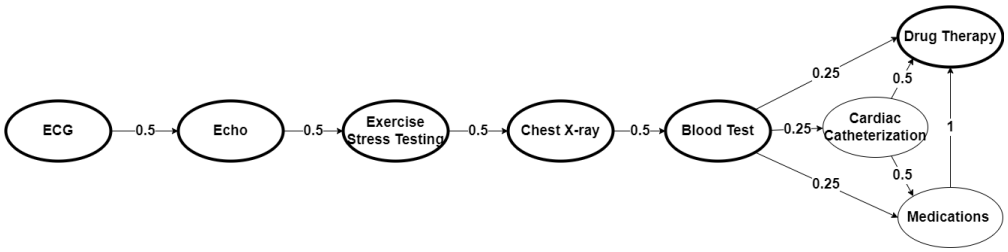
Coronary Artery Disease (CAD), Fig. 5.5c. This process is based on guidelines issued by the National Library of Medicine.[4] It is defined as a linear process of checks, cardiac catheterisation, medications, and drug therapy for patients.



(a) Advanced Practitioner Physiotherapist process



(b) Gestational Diabetes process



(c) Coronary Artery Disease process

Figure 5.5: Graphical representation of the case studies in healthcare. Figure from [11].

[4] `https://www.ncbi.nlm.nih.gov/books/NBK564304/`

Experiment Settings and Results

To evaluate the benefits of the proposed strategy for each process p in the dataset, aka its workflow, we compute its execution paths, $ExPaths(p)$, and simulate their executions. More precisely, given a process p and an execution path $ep \in ExPaths(p)$, its execution is simulated using Algorithm 2. Firstly, EC values are calculated for each task in p, and a first set of views is pre-materialised. Next, the path is traversed, which means that for each task in ep, EC values are updated, and new views are pre-materialised. As described in Algorithm 2, the simulation requires a threshold th. To use a value that is not too stringent or too relaxed, for each process p, we set a distinct th value. This is defined as the mean of EC values of p's tasks, where the task's $OPLA$ is set to 1.

To estimate the effectiveness of the strategy, we are interested in measuring the percentage of views that have been pre-materialised and actually used within ep execution (i.e., views associated with tasks in ep) as well as the percentage of views that have been pre-materialised but not used, aka wasted, during ep execution (i.e., views associated with tasks not in ep). Let us assume an execution path $ep \in ExPaths(p)$, where $|ep|$ is the number of tasks in ep, and $premat(ep)$ is the number of pre-materialised views by Algorithm 2 for tasks in ep. We compute the percentage of the pre-materialised views actually used in ep as $\%prematEP(ep) = \frac{premat(ep)}{|ep|}$. Similarly, we define the percentage of the pre-materialised views wasted in ep as $\%wastedEP(ep) = \frac{wasted(ep)}{|ep|}$, where $wasted(ep)$ is the number of views pre-materialised by Algorithm 2 and not used by tasks in ep. More in general, once all paths in $ExPaths(p)$ are simulated, we define the *average of the employed views*, Emp, as the average of $\%prematEP(ep)$, for each $ep \in ExPaths(p)$. Similarly, we define the *average of the wasted views*, Wst, as the average of $\%wastedEP(ep)$, for each $ep \in ExPaths(p)$.

The simulation platform is deployed in Python 3.10+ and is available on GitHub.[5]

Fig. 5.6 presents the results of evaluation on the BPEL benchmark dataset by varying the user types and showing each BPEL process's percentage of used (Emp) and wasted (Wst) pre-materialised views.

Overall, as reported in Fig. 5.6, the percentage of pre-materialised views is affected by OPLA values (aka user types): both Emp and Wst decline when OPLA decreases (i.e., Type4). Emp and Wst do not show linear results between clusters. For instance, clusters C1 and C2 do not have any wasted views. In cluster C3, Wst

[5] `https://github.com/GiorgiaS/hdt-viewmat.git`

is 30% and 28% for Type1 and Type2 users. Then, this value returns null in C4. Also, Emp does not have linear values from C1 to C10. These anomalies are due to the BPEL process structures and not only to the number of paths and tasks. Indeed, by manually inspecting the results, we noticed that BPEL processes where tasks appear over multiple paths have a higher Emp than those with numerous paths with few tasks in common. For instance, C9 is the cluster with the highest Wst rate. This is due to the structure of its unique process, which is characterised by numerous short paths with few tasks in common. Despite cluster C9, the other results demonstrate that the rate of Emp is always higher than Wst. Also, the difference is usually high - e.g., 94% Emp and 30% Wst for C3. In most cases, the Wst even equals 0% for all user types, as for clusters C1, C2 and C4.

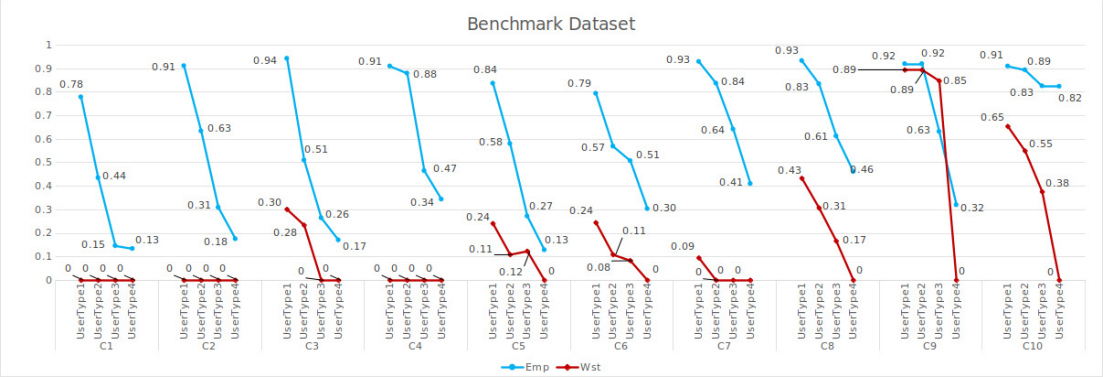


Figure 5.6: *Emp* and *Wst* by varying users' types on the BPEL benchmark dataset. Figure from [11].

The percentages of employed (Emp) and wasted (Wst) pre-materialised views for the different user types in healthcare processes are shown in Fig. 5.7. Generally, the difference between Emp and Wst is high, and Emp sometimes maintains a high value while Wst decreases. An instance is the *Tub* process for Type1, Type2 and Type3 users, where Emp always equals 100%, and Wst falls from 31% to 2%. This variation occurs because HDT-ViewMat pre-materialises fewer HDT-Views in the whole business process for Type2 and Type3 users, but those views are included in the execution path. The rate differences between processes are due to their structure and how many tasks the execution paths share. For instance, *CAD* has higher Emp than APP; indeed, tasks in the latter are traversed by fewer paths than those in the former (cfr. Fig. 5.5). Running HDT-ViewMat over those realistic processes highlights, even more than the benchmark dataset, the positive effects of this solution against wasted resources.

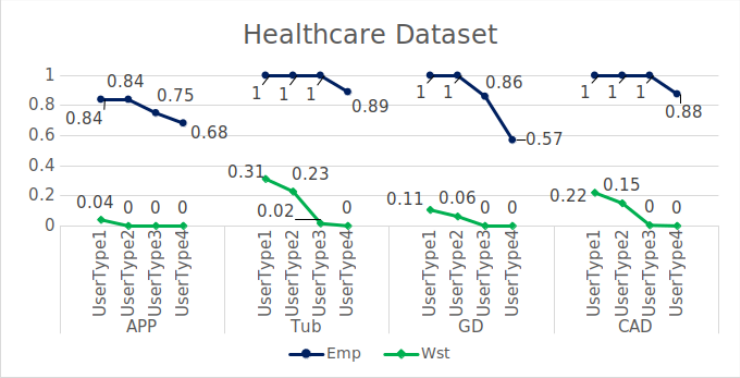


Figure 5.7: Percentage of pre-materialised HDT-Views according to the users' types on the healthcare case study dataset. Figure from [11].

To additionally prove the benefits produced by HDT-ViewMat, we assess how much the pre-materialisation reduces the waiting time for a user running a process. To estimate this gain, we assumed that each task requires a fixed amount of time δ_t to compute a view (e.g., $delta_t$= 100 units of time) and calculated the total time saved by comparing the time of execution of a path with the pre-materialisation strategy and without. In particular, without the pre-materialisation strategy, views are materialised when the tasks are invoked, thus the total waiting time of an execution path is the number of tasks multiplied by δ_t. Given a process p, we compute its average Total Time (aTT) by averaging the total waiting times of each path in $ExPaths(p)$. On the other hand, with a pre-materialisation strategy, the delay in an execution path results from the time required to materialise those views that Algorithm 2 has not yet materialised (i.e., #views not materialised multiplied by δ_t). Thus, given a process p, the average Waiting Time (aWT) is defined as the average of the waiting times of each path in $ExPaths(p)$.

Table 5.4: Saved Time for Healthcare Dataset. Table from [11].

	APP	Tub	GD	CAD
aTT	411	836	533	477
aWT	66	0	0	0

Table 5.4 and Table 5.5 show the comparison of time without (i.e., aTT) and with (i.e., aWT) the pre-materialisation strategy on the processes within the healthcare and benchmark datasets. Results in the tables refer to process execution where $OPLA = 1$ for all tasks. The benefits brought by HDT-ViewMat are evi-

dent from these tables, which demonstrate the advantages of pre-materialisation. Specifically, Table 5.4, for the healthcare dataset, highlights that the waiting time is drastically reduced for all processes. Indeed, the average time required for a user to traverse an execution path without pre-materialisation in *APP* is almost 441. In contrast, with HDT-ViewMat, this time decreases to 66. Meanwhile, the waiting time for *Tub*, *GD*, and *CAD* processes is null. Table 5.5, for the BPEL benchmark dataset, shows that our system allows users to traverse most of the execution paths with an insignificant waiting time, except C5 and C6. For processes in cluster C5, the $\overline{aWT}$ equals 29 compared to an $\overline{aTT}$ of 233, while in C6, $\overline{aWT} = 43$ and $\overline{aTT} = 510$.

Table 5.5: Saved Time for Benchmark Dataset. Table from [11].

	$C1$	$C2$	$C3$	$C4$	$C5$	$C6$	$C7$	$C8$	$C9$	$C10$
$\overline{aTT}$	275	500	99	500	233	510	521	330	300	978
$\overline{aWT}$	0	0	0	0	29	43	0	0	0	0

Overall, the results of those tests demonstrate that HDT-ViewMat allows a user to use tasks within a business process by reducing the waiting time for generating the required HDT-Views.

5.5.2 Multi-User/Provider Analysis

This section presents the experiments run to evaluate the efficiency of HDT-ViewMat in multi-user/provider scenarios. In this case, we focus on the negotiation procedure, as in the pre-materialisation, the behaviour is the same as in Section 5.5.1. We use a dataset of policies extracted from websites for this aim.

Experiment Settings and Results

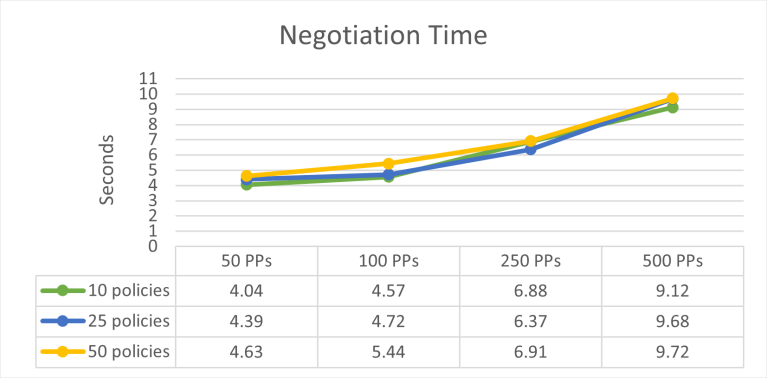
The multi-user/provider scenario introduces an additional step in the computation of ECs, namely the policy negotiation. To assess the feasibility of this component, we run a set of experiments to test the time required to determine the representative policies. As this step is mostly influenced by the complexity of the policies and preferences, we consider a different benchmark with respect to the single-user/provider scenario. In particular, we adopt the OPP-115 Corpus dataset,[6] a collection of 115 web privacy policies.

[6] `https://www.usableprivacy.org/data`

By randomly selecting values from the OPP-15 dataset, we generated several privacy policies ($GenPols$) and preferences ($GenPPs$). Specifically, to create a policy $gPol \in GenPols$, we select a purpose/third-party from a random policy in the dataset. Note that, only for the experiments' scope, we model purposes as a set (similar to the third-party component), since the OPP-115 dataset does not model purposes as a taxonomy. We believe this difference does not affect the overall time analysis, since Wu and Palmer, and Jaccard's similarity measures require similar time for their computation. Then, we set the retention period as a number in the range from 10 to 365 (days) - i.e., the smallest and biggest period identified within OPP-115.[7] Privacy preferences are generated starting from the policies in $GenPols$. Namely, to define a preference $gPP \in GenPPs$, we select a random $gPol \in GenPols$ and set gPP's purpose/third-party as a subset of the corresponding fields in $gPol$. Then, the retention period of gPP is a value below the maximum specified $GenPols$.

To solve the optimisation problem, we adopt the NSGA-III algorithm from the Pymoo library[8], due to its widespread use.

Our simulator, which is available online[9] extracts n_{pol} policies from the dataset, from which it derives n_{pp} privacy preferences. Then it computes the representative policy between n_{pol} and n_{pp} using NSGA-III for purpose and third-party fields and the average for the retention period.



	50 PPs	100 PPs	250 PPs	500 PPs
10 policies	4.04	4.57	6.88	9.12
25 policies	4.39	4.72	6.37	9.68
50 policies	4.63	5.44	6.91	9.72

Figure 5.8: Time (in seconds) required to determine the representative policy, by varying the number of policies and privacy preferences.

[7]Since the OPP-15 dataset uses terms including "stated period" or "limited" instead of explicit values, we converted them into numerical values when possible (e.g., "One calendar year" becomes 365 days).

[8]`https://pymoo.org/index.html`

[9]`https://github.com/GiorgiaS/policynegotiation.git`

We extensively test the negotiation procedure by varying the number of policies ($n_{pol} \in \{10, 25, 50\}$) and privacy preferences ($n_{pp} \in \{50, 100, 250, 500\}$). Results are shown in Fig. 5.8, which depicts the time (seconds) required to compute the representative policy. Overall, the required time grew linearly as the number of privacy preferences and policies. If we consider the worst case, our system requires less than $10sec$ to determine the representative policies and decide which views pre-materialise at the beginning of the workflow. However, this is an unusual situation when there are a lot of service providers and users. Moreover, when the users pass on to the following tasks, negotiation is no longer required, and the required time includes the EC assessment and the decision phase, which corresponds to about $0.1 \sim 0.3sec$, depending on the number of residual tasks in the execution path.

5.6 Negotiation Survey

Aiming to understand whether users would appreciate a negotiated policy, we deployed a survey on a web application, where we presented the users with the principal scenario of working space in the metaverse and asked them to select their privacy preferences under two different circumstances. The web app, developed in Python and hosted on Streamlit,[10] embodies the negotiator's logic to generate the representative policy considering the participant's privacy preferences, and then questions about the new policies and the negotiator. In particular, we asked the users to indicate their satisfaction level, plus some personal data (i.e., age and experience in IT/privacy), because we suppose these personal aspects can influence their privacy perspective.

This section presents the survey structure and the results we obtained.

5.6.1 Survey Structure

In the designed survey, participants were asked to identify themselves as an employee of an organisation called *IT-Or*, who had to join a virtual working space in the metaverse for two job meetings. To achieve this aim, all virtual space users had to share personal data, including physical appearance, with the service provider to create a realistic avatar.

Hence, the provider defined a privacy policy, the same for all job meetings, listing the purposes, the third parties, and the retention period (see Table 5.6

[10] `https://streamlit.io`

Table 5.6: Policy for both survey's scenarios

Field	Requirement
Purposes	Analytics, Additional features, Basic service, Legal requirement, Marketing, Research, Security, Service operation
Third parties	Cloud computing service, Marketing agency, Newsletter platform, Project manager, Virtual events agency, Web analytics service
Retention	90 days

for the details). The participant in the survey, in turn, had to select the privacy preferences: for purposes and third parties, the selection was a subset of the corresponding field from the policy - "Basic Service" was mandatory; meanwhile, for the retention period, the choice was a number smaller or equal to the retention period stated within the policy.

Each participant was invited to select the privacy preferences twice: one for each job meeting. We proposed two meetings that differentiated for the participant users and their preferences. For this, we simulated the users by pre-defining their preferences. These pre-defined preferences, along with the participant's own preferences and the service provider's policy, were all considered during the policy negotiation process. In particular, the two job meetings were the following:

Scenario 1. Meeting with representatives from an important customer factory. The users of this scenario were the survey participants, a manager, three peers, and two external representatives. In total, there was a policy from the service provider and seven privacy preferences, where six were pre-defined. In this case, the privacy preferences of the six simulated users were less restrictive, denoting that they were more prone to share their data.

Scenario 2. Meeting with peers. Here, the meeting was more comfortable as the users were the surveyed ones and six colleagues of the same level. This time, the six pre-defined privacy preferences of the simulated colleagues were more restrictive, denoting a smaller willingness to share users' data.

Questions proposed to the survey participants are shown in Table 5.7 and are detailed as follows:

- **Q1**: Participants had to rate their acceptance of the new policy from 1 to

Table 5.7: Saved Time for Benchmark Dataset

ID	Question	Answer	Category
Q1	To which extent would you accept the new policy compared to the initial one?	Score from 1 to 5	Scenario
Q2	Is there an element of the privacy policy that you would restrict more?	Yes/No and Multiple-choice	
Q3	If we could integrate our negotiator into a real application, would you be more likely to accept the negotiated policy than the initial one?	Score from 1 to 5	Negotiator
Q4	Do you like the negotiator?	Score from 1 to 5	
Q5	How old are you?	Single-choice	Participant
Q6	What is your level of privacy expertise?	Single-choice	

5[11] within the target scenario.

- **Q2**: Participants were asked to specify whether there was a field they would like to restrict more and, if yes, which one. As Q1, this question targeted each scenario singularly.

- **Q3**: With this question, we aim to understand whether the participants would be interested in using our negotiation mechanism in a real application scenario. They were asked to rate their preferences from 1 to 5.

- **Q4**: This question aims to assess the overall satisfaction of the negotiator, with a rate from 1 to 5.

- **Q5**: People of different age categories may have distinct concerns concerning privacy and can react differently to the negotiated policy. This question lets us comprehend whether the negotiation procedure was more attractive in some age categories.

- **Q6**: As for Q5, this question allows us to understand whether people with

[11]Rate 1 means the lower agreement/satisfaction for the user, while 5 denotes the highest.

different experiences in the IT/privacy sector show different levels of acceptance of the negotiator.

The considered questions can be grouped into three categories based on their target. Namely, Q1 and Q2 belong to the *Scenario* category, because they are related to the single scenarios; Q3 and Q4 are classified as *Negotiator*, since they ask for a comprehensive evaluation of the system; and Q5 and Q6 are included in *Participant* category, because they are related to the personality of the survey's participants.

5.6.2 Survey Partecipants

We selected 28 survey participants between the ages of 17 and 54, with any IT sector experience for work or school/university reasons. All participants were invited by sending the web app link and had to complete the survey from their devices.

First of all, the web app presented the main purpose of the survey and our research's goals. Then, it described the working space in the metaverse scenario, the data type required by the space and what participant was expected to do during the survey. After the introduction, the web app illustrated Scenario 1 and all elements of the metaverse provider's policy to the survey participants. Based on these, he/she had to select the preferences. Next, the web app computed the intermediate policy and showed the result to the participant, who had to answer questions Q1 and Q2. Subsequently, it proposed Scenario 2 to the participant similarly to Scenario 1, who he/she had to select the privacy preferences. Again, The web app computed and revealed the new policy, and the participant had to answer questions Q1 and Q2.

Finally, before storing the results, we asked the remaining questions, that is, Q3, Q4, Q5 and Q6.

5.6.3 Results Analysis

Table 5.8 illustrates the average results for questions Q1, Q2, Q3, and Q4.

As expected, the negotiated policy achieved the highest average rate in Scenario 2 than in Scenario 1, with 3.54 and 3.25, respectively. This difference is justified because the pre-defined privacy preferences were more restrictive. As a result, the representative policy was less demanding and thus, the participants were more satisfied. Regarding Scenario 1, thirteen participants specified at least one policy

Table 5.8: Average participants' answers to the survey

Questions:	Q1 Scenario 1	Q1 Scenario 2	Q3	Q4
Average results:	3.25	3.54	3.43	3.82

field on which they would have appreciated fewer requirements. Among these, 77% selected the third party, 31% the retention, and 15% the purpose fields. People with concerns about one or more fields in the representative policy in Scenario 2 were eleven; even then, the third-party field was the most targeted (63%), followed by retention (54%) and purpose (9%).

Overall, 89% of the participants expressed positive feedback[12] in accepting the representative policy compared with the initial one. Moreover, they appreciate our negotiator, since 96% of them voted positively. The average rates of 3.43 and 3.82 are favourable and denote that the audience appreciates the idea of the negotiation.

[12]We assume a score is positive if it is highest or equal to 3.

6
Conclusion

The thesis considered privacy issues in Human Digital Twins, a novel technology that is becoming increasingly successful. More specifically, an HDT is a virtual copy of an individual, which acquires data from the physical sphere (e.g., through wearable devices) and uses it to virtually model and mimic the real person. The success of this technology is the result of the features that characterise it, including the generation of a realistic virtual representation of a physical person, computations over data to predict and make decisions, monitoring of a person or a part (e.g., a system), and modelling graphical representations (e.g., 3D). Many researchers aim to take advantage of these characteristics to improve the quality of services for people (e.g., in healthcare for monitoring and treating diseases). However, the other side of using an HDT is the need for enormous personal data to maintain a reliable virtual copy of the physical person. Hence, the need for privacy solutions to protect users' data arises. An additional relevant aspect of the HDT is the time required for their materialisation, which may take minutes to hours, depending on the accuracy of the generated model.

In this thesis, we addressed the privacy requirements for HDTs by developing a proposal allowing users to state their privacy preferences based on the context. Then, we defined a strategy for faster and more efficient materialisation for complex services, which we improved with a negotiation mechanism for identifying a

representative policy in the case of multiple service providers and users. Finally, we conducted a user survey to assess user's satisfaction with using the negotiator.

In what follows, we briefly discuss how we addressed these requirements (Section 6.1), and finally, we illustrate the future works (Section 6.2).

6.1 Thesis Contributions

The first aspect we addressed was the definition of privacy preferences managing personal data collection for HDT modelling, considering that users' decisions can vary depending on the context. For this purpose, we proposed a solution, ConPrEF (Chapter 4), a context-based privacy enforcement framework for edge computing that allows users to set and enforce their contextual privacy preferences before data sharing. We designed this solution for edge computing because it can integrate HDTs to acquire, process and store the sensed data. In fact, HDTs rely on continuous data acquisition, processing, and synchronisation to accurately model the physical counterpart. Edge computing can be fundamental since edge nodes can perform these procedures, enabling data processing and analysis close to the source by reducing the latency.

To address the issue of pre-materialisation, we proposed a strategy called HDT-ViewMat (Chapter 5), which determines which views to pre-materialise based on the likelihood of task execution and user acceptance of privacy policies. Initially, we designed this strategy for a single-user/provider scenario with a single user executing a process/workflow. Then, we expand our proposal to a more challenging scenario where multiple users interact with the system simultaneously, and complex workflows involving multiple providers are executed. To handle this scenario, we enhance HDT-ViewMat with a strategy that determines a single representative policy for each task in the workflow that minimises costs for users and maximises gains for service providers. By means of HDT-ViewMat and its improved version, we provide further privacy guarantees because service providers can access only the data required for their scope (i.e., the view). In addition, the improved version allows users to share their data with fewer obligations than those stated in the initial policies, further enhancing user privacy. Then, we did tests to demonstrate the improvement - in terms of time saved by the user - introduced by our solutions and their feasibility. In the end, we deployed a web app to conduct a survey demonstrating users' appreciation for the negotiator and the representative policy.

6.2 Future Work

Starting with ConPrEF, we can make many improvements. First of all, although the PSI-CA algorithm avoids sharing private lists, it does not prevent adversaries from making inferences. For instance, when there is a single user identifier, let us say U, in the *blocklist* and the resulting cardinality is 1, it is easy to infer that U is present. A solution to this issue may be the introduction of noise or differential privacy to modify the final output, as proposed in [176]. A further issue is that the edge nodes may keep track of any connection, even if the user's preferences do not allow data sharing. This problem can be avoided through anonymous communication between edge nodes and devices. Finally, we performed the experiments in a virtual environment, simulating an individual on the move. An interesting next step could be to test our proposal in a real space, with edge nodes and people who are connected.

Concerning HDT-ViewMat, we plan to conduct a more in-depth evaluation to examine the impact of different factors on the EC metric and assess its effectiveness in predicting task execution. In doing this, we intend to consider realistic data to estimate the saved time. In fact, HDT-View materialisation can require a few minutes in the case of small amounts of data or hours if the data to be processed is large and/or the view consists of complex data representation (e.g., a high-quality virtual model). A more in-depth evaluation will also allow us to evaluate alternative similarity metrics to determine their impacts on predicting task execution. Moreover, we plan to explore the possibility of incorporating user feedback to refine the calculation of the EC metric and enhance the accuracy of predicting task execution. This could potentially lead to a more personalised and efficient workflow management system for users. Another future work is the investigation of more complex scenarios in terms of supported workflows (e.g., including parallel executions) and data types. This latter is of particular interest as supporting, for example, data with frequent updates, like continuously generated data streams (e.g., heartbeats), might require revising the strategy to pre-materialise the view. Another interesting direction is to analyse the security of the proposed framework. In particular, to avoid concerns similar to those affecting speculative execution, where attackers exploit the pre-materialised view (e.g., access a system's memory) to steal personal data.

Finally, the policy negotiation component of the improved version of HDT-ViewMat has been designed for a static scenario, where the same group of users initiate and conclude the process together. We plan to extend it to handle sit-

uations where users can leave before or join later. This dynamic setting would require continuous policy negotiation, with a consequent EC assessment, as well as the management of views that providers are already authorised to access.

List of Publications

The contributions of this thesis have been published in the following publications:

- Giorgia Sirigu, Barbara Carminati and Elena Ferrari, "Privacy and Security Issues for Human Digital Twins," in *2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA)*, 2022, pp. 1-9, doi: 10.1109/TPS-ISA56441.2022.00011.

- Giorgia Sirigu, Barbara Carminati and Elena Ferrari, "ConPrEF: A Context-based Privacy Enforcement Framework for Edge Computing," in *2023 IEEE International Conference on Edge Computing and Communications (EDGE)*, 2023, pp. 72-78, doi: 10.1109/EDGE60047.2023.00022.

- Giorgia Sirigu, Barbara Carminati, and Elena Ferrari, "Human Digital Twins: Efficient Privacy-Preserving Access Control Through Views Pre-materialisation," in *IFIP Annual Conference on Data and Applications Security and Privacy XXXVIII*, Springer, 2024, pp. 24–43, doi: 10.1007/978-3-031-65172-4_2.

- Giorgia Sirigu, Barbara Carminati, and Elena Ferrari, "Efficient Enforcement of Privacy Preferences for Human Digital Twins in Multi-User/Provider Scenarios," in *IEEE Transactions on Privacy* - under review.

The following paper has been published during my PhD but not included in this thesis:

- Ahmed Lekssays, Giorgia Sirigu, Barbara Carminati, and Elena Ferrari, "MalRec: A Blockchain-based Malware Recovery Framework for Internet of Things," in *Proceedings of the 17th International Conference on Availability, Reliability and Security*, 2022, doi: 10.1145/3538969.3544446.

Bibliography

[1] W. Shengli, "Is human digital twin possible?" *Computer Methods and Programs in Biomedicine Update*, vol. 1, 2021. DOI: 10.1016/j.cmpbup.2021.100014.

[2] "Digital twin market size, share & industry analysis, by type, by application, by enterprise type, by end-user, and regional forecast, 2024-2032," Fortune Business Insights, Tech. Rep., 2024, Report ID: FBI106246. [Online]. Available: https://www.fortunebusinessinsights.com/digital-twin-market-106246 (visited on 10/07/2024).

[3] J. Zibuschka, C. Ruff, A. Horch, and H. Roßnagel, "A human digital twin as building block of open identity management for the internet of things," in *Open Identity Summit 2020*, H. Roßnagel, C. H. Schunck, S. Mödersheim, and D. Hühnlein, Eds., Bonn: Gesellschaft für Informatik e.V., 2020, pp. 133–142. DOI: 10.18420/ois2020_11.

[4] H. Nissenbaum, "Respecting context to protect privacy: Why meaning matters," *Science and engineering ethics*, vol. 24, no. 3, pp. 831–852, 2018.

[5] G. M. Kapitsaki, "Reflecting user privacy preferences in context-aware web services," in *2013 IEEE 20th International Conference on Web Services*, 2013, pp. 123–130. DOI: 10.1109/ICWS.2013.26.

[6] H. Ou, P. Yue, Q. Duan, *et al.*, "Development of a low-cost and user-friendly system to create personalized human digital twin," in *2023 45th Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, IEEE, 2023, pp. 1–4. DOI: 10.1109/EMBC40787.2023.10340461.

[7] K. Gillette, M. A. Gsell, A. J. Prassl, *et al.*, "A framework for the generation of digital twins of cardiac electrophysiology from clinical 12-leads ecgs,"

Medical Image Analysis, vol. 71, p. 102 080, 2021, ISSN: 1361-8415. DOI: 10.1016/j.media.2021.102080.

[8] G. Sirigu, B. Carminati, and E. Ferrari, "Privacy and security issues for human digital twins," in *2022 IEEE 4th International Conference on Trust, Privacy and Security in Intelligent Systems, and Applications (TPS-ISA)*, 2022, pp. 1–9. DOI: 10.1109/TPS-ISA56441.2022.00011.

[9] K. Cao, Y. Liu, G. Meng, and Q. Sun, "An overview on edge computing research," *IEEE Access*, vol. 8, pp. 85 714–85 728, 2020. DOI: 10.1109/ACCESS.2020.2991734.

[10] G. Sirigu, B. Carminati, and E. Ferrari, "Conpref: A context-based privacy enforcement framework for edge computing," in *2023 IEEE International Conference on Edge Computing and Communications (EDGE)*, IEEE, 2023, pp. 72–78. DOI: 10.1109/EDGE60047.2023.00022.

[11] G. Sirigu, B. Carminati, and E. Ferrari, "Human digital twins: Efficient privacy-preserving access control through views pre-materialisation," in *IFIP Annual Conference on Data and Applications Security and Privacy*, Springer, 2024, pp. 24–43. DOI: 10.1007/978-3-031-65172-4_2.

[12] A. Lukács, "What is privacy? the history and definition of privacy," 2016. [Online]. Available: https://api.semanticscholar.org/CorpusID:190478776.

[13] A. Bourgeus, L. Vandercruysse, and N. Verhulst, "Understanding contextual expectations for sharing wearables' data: Insights from a vignette study," *Computers in Human Behavior Reports*, vol. 15, 2024, ISSN: 2451-9588. DOI: 10.1016/j.chbr.2024.100443.

[14] J. Colnago, L. F. Cranor, and A. Acquisti, "Is there a reverse privacy paradox? an exploratory analysis of gaps between privacy perspectives and privacy-seeking behaviors," in *Proceedings on Privacy Enhancing Technologies Symposium*, 2023, pp. 455–476. [Online]. Available: https://ssrn.com/abstract=4607259.

[15] Y. Li, "Cross-cultural privacy differences," in *Modern Socio-Technical Perspectives on Privacy*, B. P. Knijnenburg, X. Page, P. Wisniewski, H. R. Lipford, N. Proferes, and J. Romano, Eds. Cham: Springer International Publishing, 2022, pp. 267–292. DOI: 10.1007/978-3-030-82786-1_12.

[16] D. Oyserman, H. M. Coon, and M. Kemmelmeier, "Rethinking individualism and collectivism: Evaluation of theoretical assumptions and meta-analyses," *Psychological bulletin*, vol. 128, p. 3, 2002. DOI: 10.1037/0033-2909.128.1.3.

[17] H. C. Triandis, *Individualism and collectivism*. Routledge, 2018. DOI: https://doi.org/10.4324/9780429499845.

[18] T. L. James, L. Wallace, M. Warkentin, B. C. Kim, and S. E. Collignon, "Exposing others' information on online social networks (osns): Perceived shared risk, its determinants, and its influence on osn privacy control use," *Information & Management*, vol. 54, no. 7, pp. 851–865, 2017, ISSN: 0378-7206. DOI: 10.1016/j.im.2017.01.001.

[19] Y. Li, A. Kobsa, B. P. Knijnenburg, and M. C. Nguyen, "Cross-cultural privacy prediction," *Proceedings on Privacy Enhancing Technologies*, 2017. DOI: 10.1515/popets-2017-0019.

[20] M. Korir, S. Slade, W. Holmes, Y. Héliot, and B. Rienties, "Investigating the dimensions of students' privacy concern in the collection, use and sharing of data for learning analytics," *Computers in Human Behavior Reports*, vol. 9, 2023, ISSN: 2451-9588. DOI: 10.1016/j.chbr.2022.100262.

[21] E. Gratton, "If personal information is privacy's gatekeeper, then risk of harm is the key: A proposed method for determining what counts as personal information," *SSRN Electronic Journal*, Jan. 2013. DOI: 10.2139/ssrn.2334938.

[22] M. Jozani, E. Ayaburi, M. Ko, and K.-K. R. Choo, "Privacy concerns and benefits of engagement with social media-enabled apps: A privacy calculus perspective," *Computers in Human Behavior*, vol. 107, 2020, ISSN: 0747-5632. DOI: 10.1016/j.chb.2020.106260.

[23] P. of Canada, *Personal information protection and electronic documents act (pipeda)*, Accessed: 2024-11-15, 2000. [Online]. Available: https://www.priv.gc.ca/en/privacy-topics/privacy-laws-in-canada/the-personal-information-protection-and-electronic-documents-act-pipeda/.

[24] S. O. Søe, R. F. Jørgensen, and J.-E. Mai, "What is the 'personal' in 'personal information'?" *Ethics and Inf. Technol.*, vol. 23, no. 4, pp. 625–633, 2021, ISSN: 1388-1957. DOI: 10.1007/s10676-021-09600-3.

[25] N. Purtova, "The law of everything. broad concept of personal data and future of eu data protection law," *Law, Innovation and Technology*, vol. 10, no. 1, pp. 40–81, 2018. DOI: 10.1080/17579961.2018.1452176.

[26] M. Fazlioglu, "Beyond the nature of data: Obstacles to protecting sensitive information in the european union and the united states," *Fordham Urb. LJ*, vol. 46, p. 271, 2019. [Online]. Available: https://ssrn.com/abstract=4213630.

[27] C. S. Legislature, *California consumer privacy act (ccpa)*, Accessed: 2024-11-17, 2018. [Online]. Available: https://oag.ca.gov/privacy/ccpa.

[28] S. Warren and L. Brandeis, "The right to privacy," in *Killing the Messenger: 100 Years of Media Criticism*, Columbia University Press, 1989, pp. 1–21.

[29] Council of Europe, *European Convention on Human Rights (ECHR)*, Accessed: 2024-11-17, 1950. [Online]. Available: https://www.echr.coe.int/european-convention-on-human-rights.

[30] C. of Europe, *Convention for the protection of individuals with regard to automatic processing of personal data (convention 108)*, Accessed: 2024-11-17, 1991. [Online]. Available: https://www.coe.int/en/web/conventions/full-list?module=treaty-detail&treatynum=108.

[31] European Commission, *Commission Decision 2000/520/EC of 26 July 2000 on the adequacy of the protection provided by the Safe Harbor privacy principles*, Accessed: 2024-11-17, 2000. [Online]. Available: https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX%3A32000D0520.

[32] E. C. of Justice, *Judgment in case c-362/14 maximillian schrems v data protection commissioner: The court of justice declares that the commission's us safe harbour decision is invalid*, "Press Release No 117/15", Oct. 2015.

[33] E. Commission, *Commission implementing decision (eu) 2016/1250 of 12 july 2016 pursuant to directive 95/46/ec on the adequacy of the protection provided by the eu-u.s. privacy shield*, Accessed: 2024-11-17, 2016. [Online]. Available: http://data.europa.eu/eli/dec_impl/2016/1250/oj.

[34] C. of Europe, *Convention 108+ convention for the protection of individuals with regard to the processing of personal data*, Accessed: 2024-11-17, 2018. [Online]. Available: https://www.coe.int/en/web/data-protection/convention108-and-protocol.

[35] E. Commission, *Adequacy decision for the eu-u.s. data privacy framework*, Accessed: 2024-11-17, Jul. 2023. [Online]. Available: https://ec.europa.eu/commission/presscorner/detail/en/ip_23_3721.

[36] European Parliament and Council of the European Union, *General data protection regulation (gdpr)*, accessed: 2024-11-17, May 4, 2016. [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[37] B. R. Barricelli, E. Casiraghi, and D. Fogli, "A survey on digital twin: Definitions, characteristics, applications, and design implications," *IEEE Access*, vol. 7, pp. 167 653–167 671, 2019. DOI: 10.1109/ACCESS.2019.2953499.

[38] S. Khan, T. Arslan, and T. Ratnarajah, "Digital twin perspective of fourth industrial and healthcare revolution," *IEEE Access*, vol. 10, pp. 25 732–25 754, 2022. DOI: 10.1109/ACCESS.2022.3156062.

[39] C. Alcaraz and J. Lopez, "Digital twin: A comprehensive survey of security threats," *IEEE Communications Surveys & Tutorials*, vol. 24, no. 3, pp. 1475–1503, 2022. DOI: 10.1109/COMST.2022.3171465.

[40] AnyLogic, "An introduction to digital twin development," The AnyLogic Company, Tech. Rep., 2018. [Online]. Available: https://www.anylogic.com/resources/white-papers/an-introduction-to-digital-twin-development/.

[41] M. Sprinzen, "Digital twins will drive the future of digital transformation," VANTIQ Inc, Tech. Rep., 2020. [Online]. Available: https://www.edgeir.com/digital-twins-will-drive-the-future-of-digital-transformation-20210822.

[42] G. Healthcare, "Capacity strategy powered by a digital twin execute briefing," General Electric Company, Tech. Rep., 2020. [Online]. Available: https://www.gehccommandcenter.com/digital-twin.

[43] E. Karaarslan and M. Babiker, "Digital twin security threats and countermeasures: An introduction," in *2021 International Conference on Information Security and Cryptology*, 2021, pp. 7–11. DOI: 10.1109/ISCTURKEY53027.2021.9654360.

[44] M. Grieves and J. Vickers, "Digital twin: Mitigating unpredictable, undesirable emergent behavior in complex systems," in *Transdisciplinary Perspectives on Complex Systems: New Findings and Approaches*, F.-J. Kahlen, S. Flumerfelt, and A. Alves, Eds. Cham: Springer International Publishing, 2017, pp. 85–113, ISBN: 978-3-319-38756-7. DOI: 10.1007/978-3-319-38756-7_4.

[45] M. Segovia and J. Garcia-Alfaro, "Design, modeling and implementation of digital twins," *Sensors*, vol. 22, no. 14, 2022. DOI: 10.3390/s22145396.

[46] M. van Dyck, D. Lüttgens, F. T. Piller, and S. Brenk, "Interconnected digital twins and the future of digital manufacturing: Insights from a delphi study," *Journal of Product Innovation Management*, vol. 40, no. 4, pp. 475–505, 2023. DOI: 10.1111/jpim.12685. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/jpim.12685.

[47] J. Jagannath, K. Ramezanpour, and A. Jagannath, "Digital twin virtualization with machine learning for iot and beyond 5g networks: Research directions for security and optimal control," in *Proceedings of the 2022 ACM Workshop on Wireless Security and Machine Learning*, ser. WiseML '22, San Antonio, TX, USA: Association for Computing Machinery, 2022, pp. 81–86, ISBN: 9781450392778. DOI: 10.1145/3522783.3529519.

[48] D. Holmes, M. Papathanasaki, L. Maglaras, M. A. Ferrag, S. Nepal, and H. Janicke, "Digital twins and cyber security – solution or challenge?" In *2021 6th South-East Europe Design Automation, Computer Engineering, Computer Networks and Social Media Conference (SEEDA-CECNSM)*, 2021, pp. 1–8. DOI: 10.1109/SEEDA-CECNSM53056.2021.9566277.

[49] M. Eckhart and A. Ekelhart, "Towards security-aware virtual environments for digital twins," in *Proceedings of the 4th ACM Workshop on Cyber-Physical System Security*, ser. CPSS '18, Incheon, Republic of Korea: Association for Computing Machinery, 2018, pp. 61–72, ISBN: 9781450357555. DOI: 10.1145/3198458.3198464.

[50] F. Pires, A. Cachada, J. Barbosa, A. P. Moreira, and P. Leitão, "Digital twin in industry 4.0: Technologies, applications and challenges," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, vol. 1, 2019, pp. 721–726. DOI: 10.1109/INDIN41052.2019.8972134.

[51] J. C. Olivares-Rojas, E. Reyes-Archundia, J. A. Gutiérrez-Gnecchi, I. Molina-Moreno, J. Cerda-Jacobo, and A. Méndez-Patiño, "Towards cybersecurity of the smart grid using digital twins," *IEEE Internet Computing*, vol. 26, no. 3, pp. 52–57, 2022. DOI: 10.1109/MIC.2021.3063674.

[52] O. Veledar, V. Damjanovic-Behrendt, and G. Macher, "Digital twins for dependability improvement of autonomous driving," in *Systems, Software and Services Process Improvement*, A. Walker, R. V. O'Connor, and R. Messnarz, Eds., Cham: Springer International Publishing, 2019, pp. 415–426.

[53] V. Damjanovic-Behrendt, "A digital twin-based privacy enhancement mechanism for the automotive industry," in *2018 International Conference on Intelligent Systems (IS)*, 2018, pp. 272–279. DOI: 10.1109/IS.2018.8710526.

[54] M. E. Miller and E. Spatz, "A unified view of a human digital twin," *Human-Intelligent Systems Integration*, pp. 1–11, 2022. DOI: 10.1007/s42454-022-00041-x.

[55] A. Löcklin, T. Jung, N. Jazdi, T. Ruppert, and M. Weyrich, "Architecture of a human-digital twin as common interface for operator 4.0 applications," *Procedia CIRP*, vol. 104, pp. 458–463, Jan. 2021. DOI: 10.1016/j.procir.2021.11.077.

[56] S. Sengan, K. Kumar, V. Subramaniyaswamy, and L. Ravi, "Cost-effective and efficient 3d human model creation and re-identification application for human digital twins," *Multimedia Tools and Applications*, pp. 1–18, 2022. DOI: 10.1007/s11042-021-10842-y.

[57] B. Wang, H. Zhou, X. Li, *et al.*, "Human digital twin in the context of industry 5.0," *Robotics and Computer-Integrated Manufacturing*, vol. 85, p. 102 626, 2024, ISSN: 0736-5845. DOI: 10.1016/j.rcim.2023.102626.

[58] Z. Johnson and M. J. Saikia, "Digital twins for healthcare using wearables," *Bioengineering*, vol. 11, no. 6, 2024, ISSN: 2306-5354. DOI: 10.3390/bioengineering11060606.

[59] Y. Lin, L. Chen, A. Ali, *et al.*, "Human digital twin: A survey," *Journal of Cloud Computing*, vol. 13, no. 1, 2024. DOI: 10.1186/s13677-024-00691-z.

[60] M. Alazab, L. Khan, S. Koppu, *et al.*, "Digital twins for healthcare 4.0: Recent advances, architecture, and open challenges," *IEEE Consumer Electronics Magazine*, vol. 12, no. 6, pp. 29–37, Nov. 2023, ISSN: 2162-2248. DOI: 10.1109/MCE.2022.3208986.

[61] B. R. Barricelli, E. Casiraghi, J. Gliozzo, A. Petrini, and S. Valtolina, "Human digital twin for fitness management," *IEEE Access*, vol. 8, pp. 26 637–26 664, 2020. DOI: 10.1109/ACCESS.2020.2971576.

[62] N. Bagaria, F. Laamarti, H. F. Badawi, A. Albraikan, R. A. Martinez Velazquez, and A. El Saddik, "Health 4.0: Digital twins for health and wellbeing," in *Connected Health in Smart Cities*, A. El Saddik, M. S. Hossain, and B. Kantarci, Eds. Cham: Springer International Publishing, 2020, pp. 143–152, ISBN: 978-3-030-27844-1. DOI: 10.1007/978-3-030-27844-1_7.

[63] H. Elayan, M. Aloqaily, and M. Guizani, "Digital twin for intelligent context-aware iot healthcare systems," *IEEE Internet of Things Journal*, vol. 8, no. 23, pp. 16 749–16 757, 2021. DOI: 10.1109/JIOT.2021.3051158.

[64] J. Corral-Acero, F. Margara, M. Marciniak, *et al.*, "The 'Digital Twin' to enable the vision of precision cardiology," *European Heart Journal*, vol. 41, no. 48, pp. 4556–4564, Mar. 2020, ISSN: 0195-668X. DOI: 10.1093/eurheartj/ehaa159.

[65] N. K. Chakshu, I. Sazonov, and P. Nithiarasu, "Towards enabling a cardiovascular digital twin for human systemic circulation using inverse analysis," *Biomechanics and modeling in mechanobiology*, vol. 20, no. 2, pp. 449–465, 2021. DOI: 10.1007/s10237-020-01393-6.

[66] N. K. Chakshu, J. Carson, I. Sazonov, and P. Nithiarasu, "A semi-active human digital twin model for detecting severity of carotid stenoses from head vibration - a coupled computational mechanics and computer vision method," *International Journal for Numerical Methods in Biomedical Engineering*, vol. 35, Jan. 2019. DOI: 10.1002/cnm.3180.

[67] I. Voigt, H. Inojosa, A. Dillenseger, R. Haase, K. Akgün, and T. Ziemssen, "Digital twins for multiple sclerosis," *Frontiers in immunology*, vol. 12, 2021. DOI: 10.3389/fimmu.2021.669811.

[68] D. Mourtzis, J. Angelopoulos, N. Panopoulos, and D. Kardamakis, "A smart iot platform for oncology patient diagnosis based on ai: Towards the human digital twin," *Procedia CIRP*, vol. 104, pp. 1686–1691, 2021. DOI: 10.1016/j.procir.2021.11.284.

[69] B. Björnsson, C. Borrebaeck, N. Elander, *et al.*, "Digital twins to personalize medicine," *Genome medicine*, vol. 12, no. 1, pp. 1–4, 2020. DOI: 10.1186/s13073-019-0701-3.

[70] K. Subramanian, "Digital twin for drug discovery and development—the virtual liver," *Journal of the Indian Institute of Science*, vol. 100, no. 4, pp. 653–662, 2020. DOI: 10.1007/s41745-020-00185-2.

[71] J. Chen, C. Yi, H. Du, *et al.*, "A revolution of personalized healthcare: Enabling human digital twin with mobile aigc," *IEEE Network*, pp. 1–1, 2024. DOI: 10.1109/MNET.2024.3366560.

[72] R. Gámez Díaz, Q. Yu, Y. Ding, F. Laamarti, and A. El Saddik, "Digital twin coaching for physical activities: A survey," *Sensors*, vol. 20, no. 20, 2020, ISSN: 1424-8220. [Online]. Available: https://www.mdpi.com/1424-8220/20/20/5936.

[73] M. W. Lauer-Schmaltz, P. Cash, J. P. Hansen, and A. Maier, "Designing human digital twins for behaviour-changing therapy and rehabilitation: A systematic review," *Proceedings of the Design Society*, vol. 2, pp. 1303–1312, 2022. DOI: 10.1017/pds.2022.132.

[74] D. Romero, P. Bernus, O. Noran, J. Stahre, and Å. Fast-Berglund, "The operator 4.0: Human cyber-physical systems & adaptive automation towards human-automation symbiosis work systems," in *Advances in Production Management Systems. Initiatives for a Sustainable World*, I. Nääs, O. Vendrametto, J. Mendes Reis, *et al.*, Eds., Cham: Springer International Publishing, 2016, pp. 677–686, ISBN: 978-3-319-51133-7. DOI: 10.1007/978-3-319-51133-7_80.

[75] C. M. Mitchell, "Human-centered automation: A philosophy, some design tenets, and related research," in *Human Interaction with Complex Systems: Conceptual Principles and Design Practice*. Boston, MA: Springer US, 1996, pp. 377–381, ISBN: 978-1-4613-1447-9. DOI: 10.1007/978-1-4613-1447-9_31.

[76] E. Kaasinen, F. Schmalfuß, C. Özturk, *et al.*, "Empowering and engaging industrial workers with operator 4.0 solutions," *Computers & Industrial Engineering*, vol. 139, p. 105 678, 2020, ISSN: 0360-8352. DOI: 10.1016/j.cie.2019.01.052.

[77] E. Commission, D.-G. for Research, Innovation, M. Breque, L. De Nul, and A. Petridis, *Industry 5.0: towards a sustainable, human-centric and resilient European industry*. Publications Office of the European Union, 2021. DOI: https://doi.org/10.2777/308407.

[78] E. Montini, N. Bonomi, F. Daniele, *et al.*, "The human-digital twin in the manufacturing industry: Current perspectives and a glimpse of future," in *Trusted Artificial Intelligence in Manufacturing: A Review of the Emerging Wave of Ethical and Human Centric AI Technologies for Smart Production*, J. Soldatos and D. Kyriazis, Eds. Now Publishers, Sep. 2021, pp. 132–147, ISBN: 978-1-68083-876-3. DOI: 10.1561/9781680838770.ch7.

[79] X. Li, A. Nassehi, B. Wang, S. J. Hu, and B. I. Epureanu, "Human-centric manufacturing for human-system coevolution in industry 5.0," *CIRP Annals*, vol. 72, no. 1, pp. 393–396, 2023, ISSN: 0007-8506. DOI: 10.1016/j.cirp.2023.04.039.

[80] M. Fera, A. Greco, M. Caterino, *et al.*, "Towards digital twin implementation for assessing production line performance and balancing," *Sensors*, vol. 20, no. 1, 2020, ISSN: 1424-8220. DOI: 10.3390/s20010097.

[81] I. Graessler and A. Poehler, "Integration of a digital twin as human representation in a scheduling procedure of a cyber-physical production system," in *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2017, pp. 289–293. DOI: 10.1109/IEEM.2017.8289898.

[82] I. Graessler and A. Poehler, "Intelligent control of an assembly station by integration of a digital twin for employees into the decentralized control system," *Procedia Manufacturing*, vol. 24, pp. 185–189, 2018, 4th International Conference on System-Integrated Intelligence: Intelligent, Flexible and Connected Systems in Products and Production, ISSN: 2351-9789. DOI: 10.1016/j.promfg.2018.06.041.

[83] A. Greco, M. Caterino, M. Fera, and S. Gerbino, "Digital twin for monitoring ergonomics during manufacturing production," *Applied Sciences*, vol. 10, no. 21, 2020, ISSN: 2076-3417. DOI: 10.3390/app10217758.

[84] F. Caputo, A. Greco, M. Fera, and R. Macchiaroli, "Digital twins to enhance the integration of ergonomics in the workplace design," *International Journal of Industrial Ergonomics*, vol. 71, pp. 20–31, 2019, ISSN: 0169-8141. DOI: 10.1016/j.ergon.2019.02.001.

[85] S. Baskaran, F. A. Niaki, M. Tomaszewski, *et al.*, "Digital human and robot simulation in automotive assembly using siemens process simulate: A feasibility study," *Procedia Manufacturing*, vol. 34, pp. 986–994, 2019, 47th SME North American Manufacturing Research Conference, NAMRC 47, Pennsylvania, USA., ISSN: 2351-9789. DOI: 10.1016/j.promfg.2019.06.097.

[86] N. Nikolakis, K. Alexopoulos, E. Xanthakis, and G. Chryssolouris, "The digital twin implementation for linking the virtual representation of human-based production tasks to their physical counterpart in the factory-floor," *International Journal of Computer Integrated Manufacturing*, vol. 32, no. 1, pp. 1–12, Jan. 2019. DOI: 10.1080/0951192X.2018.1529430.

[87] A. Bilberg and A. A. Malik, "Digital twin driven human–robot collaborative assembly," *CIRP Annals*, vol. 68, no. 1, pp. 499–502, 2019, ISSN: 0007-8506. DOI: 10.1016/j.cirp.2019.04.011.

[88] C. Constantinescu, R. Rus, C.-A. Rusu, and D. Popescu, "Digital twins of exoskeleton-centered workplaces: Challenges and development methodology," *Procedia Manufacturing*, vol. 39, pp. 58–65, 2019, 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing August 9-14, 2019 — Chicago, Illinois (USA), ISSN: 2351-9789. DOI: 10.1016/j.promfg.2020.01.228.

[89] P. Zheng, S. Li, J. Fan, C. Li, and L. Wang, "A collaborative intelligence-based approach for handling human-robot collaboration uncertainties," *CIRP Annals*, vol. 72, no. 1, pp. 1–4, 2023, ISSN: 0007-8506. DOI: 10.1016/j.cirp.2023.04.057.

[90] P. Saariluoma, A. Karvonen, and L. Sorsamäki, "Human digital twins in acquiring information about human mental processes for cognitive mimetics," in Jan. 2022, ISBN: 9781643682426. DOI: 10.3233/FAIA210484.

[91] R. Di Pietro and S. Cresci, "Metaverse: Security and privacy issues," in *2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA)*, 2021, pp. 281–288. DOI: 10.1109/TPSISA52974.2021.00032.

[92] H. Ning, H. Wang, Y. Lin, *et al.*, "A survey on metaverse: The state-of-the-art, technologies, applications, and challenges," *arXiv preprint arXiv:2111.09673*, 2021. DOI: 10.48550/ARXIV.2111.09673.

[93] S. Banaeian Far and A. Imani Rad, "Applying digital twins in metaverse: User interface, security and privacy challenges," *Journal of Metaverse*, vol. 2, pp. 8–16, Apr. 2022.

[94] P. Ruiu, M. Nitti, V. Pilloni, M. Cadoni, E. Grosso, and M. Fadda, "Metaverse & human digital twin: Digital identity, biometrics, and privacy in the future virtual worlds," *Multimodal Technologies and Interaction*, vol. 8, no. 6, 2024, ISSN: 2414-4088. DOI: 10.3390/mti8060048.

[95] Gartner. "Gartner predicts 25% of people will spend at least one hour per day in the metaverse by 2026." (Feb. 7, 2024), [Online]. Available: https://www.gartner.com/en/newsroom/press-releases/2022-02-07-gartner-predicts-25-percent-of-people-will-spend-at-least-one-hour-per-day-in-the-metaverse-by-2026. (accessed: 2024-10-30).

[96] M. Hearn and S. Rix, "Cybersecurity considerations for digital twin implementations," *IIC Journal of Innovation*, pp. 107–113, 2019.

[97] A. J. G. de Azambuja, T. Giese, K. Schützer, R. Anderl, B. Schleich, and V. R. Almeida, "Digital twins in industry 4.0 – opportunities and challenges related to cyber security," *Procedia CIRP*, vol. 121, pp. 25–30, 2024, 11th CIRP Global Web Conference (CIRPe 2023), ISSN: 2212-8271. DOI: 10.1016/j.procir.2023.09.225.

[98] Y. Wang, Z. Su, S. Guo, M. Dai, T. H. Luan, and Y. Liu, "A survey on digital twins: Architecture, enabling technologies, security and privacy, and future prospects," *IEEE Internet of Things Journal*, vol. 10, no. 17, pp. 14 965–14 987, 2023. DOI: 10.1109/JIOT.2023.3263909.

[99] A. Abdullah, R. Hamad, M. Abdulrahman, H. Moala, and S. Elkhediri, "CyberSecurity: A review of internet of things (IoT) security issues, challenges and techniques," in *2019 2nd International Conference on Computer Applications & Information Security (ICCAIS)*, IEEE, May 2019, pp. 1–6. DOI: 10.1109/CAIS.2019.8769560.

[100] E. Pärn, N. Ghadiminia, B. García de Soto, and K. Oti-Sarpong, "A perfect storm: Digital twins, cybersecurity, and general contracting firms," *Developments in the Built Environment*, vol. 18, p. 100 466, 2024, ISSN: 2666-1659. DOI: 10.1016/j.dibe.2024.100466.

[101] W. Syafitri, Z. Shukur, U. A. Mokhtar, R. Sulaiman, and M. A. Ibrahim, "Social engineering attacks prevention: A systematic literature review," *IEEE Access*, vol. 10, pp. 39 325–39 343, 2022. DOI: 10.1109/ACCESS.2022.3162594.

[102] F. Akbarian, E. Fitzgerald, and M. Kihl, "Intrusion detection in digital twins for industrial control systems," in *2020 International Conference on Software, Telecommunications and Computer Networks (SoftCOM)*, 2020, pp. 1–6. DOI: 10.23919/SoftCOM50211.2020.9238162.

[103] G. Sugumar and A. Mathur, "Assessment of a method for detecting process anomalies using digital-twinning," in *2019 15th European Dependable Computing Conference (EDCC)*, 2019, pp. 119–126. DOI: 10.1109/EDCC.2019.00031.

[104] S. Abdullahi, A. Zare, and S. Lazarova-Molnar, "Cybersecurity in distributed industrial digital twins: Threats, defenses, and key takeaways," in *The 1st International Workshop on Distributed Digital Twins (DiDiT2024)*, Jun. 2024.

[105] S. Suhail, R. Jurdak, R. Hussain, and D. Svetinovic, "Security attacks and solutions for digital twins," arXiv, 2022. DOI: 10.48550/ARXIV.2202.12501.

[106] J. I. Jimenez, H. Jahankhani, and S. Kendzierskyj, "Health care in the cyberspace: Medical cyber-physical system and digital twin challenges," in *Digital Twin Technologies and Smart Cities*, M. Farsi, A. Daneshkhah, A. Hosseinian-Far, and H. Jahankhani, Eds. Cham: Springer International Publishing, 2020, pp. 79–92, ISBN: 978-3-030-18732-3. DOI: 10.1007/978-3-030-18732-3_6.

[107] B. C. Singh, B. Carminati, and E. Ferrari, "Privacy-aware personal data storage (p-pds): Learning how to protect user privacy from external applications," *IEEE Transactions on Dependable and Secure Computing*, vol. 18, no. 2, pp. 889–903, 2021. DOI: 10.1109/TDSC.2019.2903802.

[108] K. Bruynseels, F. Santoni de Sio, and J. van den Hoven, "Digital twins in health care: Ethical implications of an emerging engineering paradigm," *Frontiers in genetics*, vol. 9, p. 31, Feb. 2018. DOI: 10.3389/fgene.2018.00031.

[109] C. De Maeyer and P. Markopoulos, "Are digital twins becoming our personal (predictive) advisors?: 'our digital mirror of who we were, who we are and who we will become'," in Jul. 2020, pp. 250–268, ISBN: 978-3-030-50248-5. DOI: 10.1007/978-3-030-50249-2_19.

[110] S. Lipsa and R. Dash, "A novel intrusion detection system based on deep learning and random forest for digital twin on iot platform," vol. 2, pp. 51–64, Mar. 2023. DOI: 10.56781/ijsret.2023.2.1.0020.

[111] E. Schiller, A. Aidoo, J. Fuhrer, J. Stahl, M. Ziörjen, and B. Stiller, "Landscape of iot security," *Computer Science Review*, vol. 44, 2022, ISSN: 1574-0137. DOI: 10.1016/j.cosrev.2022.100467.

[112] T. Li, H. Wang, D. He, and J. Yu, "Synchronized provable data possession based on blockchain for digital twin," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 472–485, 2022. DOI: 10.1109/TIFS.2022.3144869.

[113] G. Cathey, J. Benson, M. Gupta, and R. Sandhu, "Edge centric secure data sharing with digital twins in smart ecosystems," in *IEEE Int. Conf. TPS-ISA*, USA: IEEE, 2021, pp. 70–79.

[114] S. Amofa, Q. Xia, H. Xia, *et al.*, "Blockchain-secure patient digital twin in healthcare using smart contracts," *PLOS ONE*, vol. 19, no. 2, pp. 1–28, Feb. 2024. DOI: 10.1371/journal.pone.0286120.

[115] A. Khan, F. Shahid, C. Maple, A. Ahmad, and G. Jeon, "Toward smart manufacturing using spiral digital twin framework and twinchain," *IEEE Transactions on Industrial Informatics*, vol. PP, pp. 1–1, Dec. 2020. DOI: 10.1109/TII.2020.3047840.

[116] S. Liao, J. Wu, A. K. Bashir, W. Yang, J. Li, and U. Tariq, "Digital twin consensus for blockchain-enabled intelligent transportation systems in smart cities," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 22 619–22 629, 2022. DOI: 10.1109/TITS.2021.3134002.

[117] Q. Liu, P. Li, W. Zhao, W. Cai, S. Yu, and V. C. M. Leung, "A survey on security threats and defensive techniques of machine learning: A data driven view," *IEEE Access*, vol. 6, pp. 12 103–12 117, 2018. DOI: 10.1109/ACCESS.2018.2805680.

[118] S. Banabilah, M. Aloqaily, E. Alsayed, N. Malik, and Y. Jararweh, "Federated learning review: Fundamentals, enabling technologies, and future applications," *Information Processing & Management*, vol. 59, no. 6, 2022, ISSN: 0306-4573. DOI: 10.1016/j.ipm.2022.103061.

[119] Z. Lv, C. Cheng, and H. Lv, "Blockchain-based decentralized learning for security in digital twins," *IEEE Internet of Things Journal*, vol. 10, no. 24, pp. 21 479–21 488, 2023. DOI: 10.1109/JIOT.2023.3295499.

[120] M. Dietz, B. Putz, and G. Pernul, "A distributed ledger approach to digital twin secure data sharing," in *Data and Application Security and Privacy XXXIII*, ser. LNCS, vol. 11559, Cham: Springer International Publishing, 2019, pp. 281–300.

[121] B. Putz, M. Dietz, P. Empl, and G. Pernul, "Ethertwin: Blockchain-based secure digital twin information management," *Information Processing & Management*, vol. 58, no. 1, p. 102 425, 2021, ISSN: 0306-4573. DOI: 10.1016/j.ipm.2020.102425.

[122] P. Bellavista, C. Giannelli, M. Mamei, M. Mendula, and M. Picone, "Digital twin oriented architecture for secure and qos aware intelligent communications in industrial environments," *Pervasive and Mobile Computing*, vol. 85, p. 101 646, 2022, ISSN: 1574-1192. DOI: 10.1016/j.pmcj.2022.101646.

[123] H. Xiong, Z. Qu, X. Huang, and K.-H. Yeh, "Revocable and unbounded attribute-based encryption scheme with adaptive security for integrating digital twins in internet of things," *IEEE Journal on Selected Areas in Communications*, vol. 41, no. 10, pp. 3306–3317, 2023. DOI: 10.1109/JSAC.2023.3310076.

[124] A. De Benedictis, C. Esposito, and A. Somma, "Toward the adoption of secure cyber digital twins to enhance cyber-physical systems security," in *Quality of Information and Communications Technology*, A. Vallecillo, J. Visser, and R. Pérez-Castillo, Eds., Cham: Springer International Publishing, 2022, pp. 307–321. DOI: 10.1007/978-3-031-14179-9_21.

[125] O. Ulusoy and P. Yolum, "Emergent privacy norms for collaborative systems," in *PRIMA 2019: Principles and Practice of Multi-Agent Systems*, Springer, 2019, pp. 514–522. DOI: 10.1007/978-3-030-33792-6_36.

[126] F. Mosca and J. M. Such, "Elvira: An explainable agent for value and utility-driven multiuser privacy," in *Proceedings of the 20th International Conference on Autonomous Agents and MultiAgent Systems*, 2021, pp. 916–924, ISBN: 9781450383073.

[127] J. M. Such and M. Rovatsos, "Privacy policy negotiation in social media," *ACM Trans. Auton. Adapt. Syst.*, vol. 11, no. 1, Feb. 2016. DOI: 10.1145/2821512.

[128] K. Kursawe, G. Neven, and P. Tuyls, "Private policy negotiation," in *Financial Cryptography and Data Security*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 81–95, ISBN: 978-3-540-46256-9. DOI: 10.1007/11889663_6.

[129] O. R. Sanchez, I. Torre, and B. P. Knijnenburg, "Semantic-based privacy settings negotiation and management," *FGCS*, vol. 111, pp. 879–898, 2020. DOI: 10.1016/j.future.2019.10.024.

[130] D. D. Walker, E. G. Mercer, and K. E. Seamons, "Or best offer: A privacy policy negotiation protocol," in *2008 IEEE Workshop on Policies for Distributed Systems and Networks*, IEEE, 2008, pp. 173–180. DOI: 10.1109/POLICY.2008.39.

[131] K. Alanezi and S. Mishra, "Incorporating individual and group privacy preferences in the internet of things," *Ambient Intell. Humaniz. Comput.*, vol. 13, no. 4, pp. 1969–1984, 2022. DOI: 10.1007/s12652-021-02959-7.

[132] P. Ranaweera, A. D. Jurcut, and M. Liyanage, "Survey on multi-access edge computing security and privacy," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 2, pp. 1078–1124, 2021. DOI: 10.1109/COMST.2021.3062546.

[133] J. Tian, Y. Wang, and Y. Shen, "An identity-based authentication scheme with full anonymity and unlinkability for mobile edge computing," *IEEE Internet of Things Journal*, vol. 11, no. 13, pp. 23 561–23 576, 2024. DOI: 10.1109/JIOT.2024.3385095.

[134] S. A. Soleymani, S. Goudarzi, M. H. Anisi, A. Jindal, N. Kama, and S. A. Ismail, "A privacy-preserving authentication scheme for real-time medical monitoring systems," *IEEE Journal of Biomedical and Health Informatics*, vol. 27, no. 5, pp. 2314–2322, 2023. DOI: 10.1109/JBHI.2022.3143207.

[135] Y. Li, Q. Cheng, X. Liu, and X. Li, "A secure anonymous identity-based scheme in new authentication architecture for mobile edge computing," *IEEE Systems Journal*, vol. 15, no. 1, pp. 935–946, 2020. DOI: 10.1109/JSYST.2020.2979006.

[136] Y. Wang, X. Jia, Y. Xia, M. K. Khan, and D. He, "A blockchain-based conditional privacy-preserving authentication scheme for edge computing services," *Journal of Information Security and Applications*, vol. 70, p. 103 334, 2022, ISSN: 2214-2126. DOI: 10.1016/j.jisa.2022.103334.

[137] J. Kang, X. Luo, J. Nie, *et al.*, "Blockchain-based pseudonym management for vehicle twin migrations in vehicular edge metaverse," *IEEE Internet of Things Journal*, 2024. DOI: 10.1109/JIOT.2024.3404559.

[138] F. Buccafurri, V. De Angelis, M. F. Idone, and C. Labrini, "A hierarchical distributed trusted location service achieving location k-anonymity against the global observer," *Computer Networks*, vol. 243, 2024. DOI: 10.1016/j.comnet.2024.110301.

[139] Y. Wang, Z. Tian, S. Su, Y. Sun, and C. Zhu, "Preserving location privacy in mobile edge computing," in *ICC 2019-2019 IEEE International Conference on Communications (ICC)*, IEEE, 2019, pp. 1–6. DOI: 10.1109/ICC.2019.8761370.

[140] P. Zhang, M. Durresi, and A. Durresi, "Internet network location privacy protection with multi-access edge computing," *Computing*, vol. 103, pp. 473–490, 2021. DOI: 10.1007/s00607-020-00860-3.

[141] S. Zeng, H. Zhang, F. Hao, and H. Li, "Deniable-based privacy-preserving authentication against location leakage in edge computing," *IEEE Systems Journal*, vol. 16, no. 2, pp. 1729–1738, 2022. DOI: 10.1109/JSYST.2021.3049629.

[142] G. Cui, Q. He, F. Chen, H. Jin, Y. Xiang, and Y. Yang, "Location privacy protection via delocalization in 5g mobile edge computing environment," *IEEE Transactions on Services Computing*, vol. 16, no. 1, pp. 412–423, 2023. DOI: 10.1109/TSC.2021.3112659.

[143] M. Bi, Y. Wang, Z. Cai, and X. Tong, "A privacy-preserving mechanism based on local differential privacy in edge computing," *China Communications*, vol. 17, no. 9, pp. 50–65, 2020. DOI: 10.23919/JCC.2020.09.005.

[144] G. Zhang, J. Du, X. Yuan, and K. Zhang, "Differential privacy-based location privacy protection for edge computing networks," *Electronics*, vol. 13, p. 3510, Sep. 2024. DOI: 10.3390/electronics13173510.

[145] Z. Wang, Y. Sun, D. Liu, *et al.*, "Location privacy-aware task offloading in mobile edge computing," *IEEE Transactions on Mobile Computing*, vol. 23, no. 3, pp. 2269–2283, 2024. DOI: 10.1109/TMC.2023.3254553.

[146] G. Zhang, S. Zhang, Z. Man, C. Cui, and W. Hu, "Location privacy protection in edge computing: Co-design of differential privacy and offloading mode," *Electronics*, vol. 13, no. 13, 2024. DOI: 10.3390/electronics13132668.

[147] B. Yang, A. Liu, N. N. Xiong, T. Wang, and S. Zhang, "Vlr-bpp: An intelligent virtual location replacement based bilateral privacy-preserving architecture for edge cloud systems," *Future Generation Computer Systems*, vol. 163, p. 107 488, 2025, ISSN: 0167-739X. DOI: 10.1016/j.future.2024.107488.

[148] D. C. Nguyen, P. N. Pathirana, M. Ding, and A. Seneviratne, "A cooperative architecture of data offloading and sharing for smart healthcare with blockchain," in *2021 IEEE International Conference on Blockchain and Cryptocurrency (ICBC)*, 2021, pp. 1–8. DOI: 10.1109/ICBC51069.2021.9461063.

[149] L. Zhang, M. Peng, W. Wang, Z. Jin, Y. Su, and H. Chen, "Secure and efficient data storage and sharing scheme for blockchain-based mobile-edge computing," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 10, Oct. 2021, ISSN: 2161-3915. DOI: 10.1002/ett.4315.

[150] H. Zhang, L. Cao, N. Kumar, J. Zhang, P. Zhang, and J. Wang, "An improved ddpg-based privacy sensitive level protection computation offloading method in mobile edge computing," *Future Generation Computer Systems*, vol. 159, pp. 522–532, 2024, ISSN: 0167-739X. DOI: 10.1016/j.future.2024.05.018.

[151] C. Lachner, T. Rausch, and S. Dustdar, "Context-aware enforcement of privacy policies in edge computing," in *2019 IEEE International Congress on Big Data (BigDataCongress)*, Milan, Italy: IEEE Computer Society, 2019, pp. 1–6. DOI: 10.1109/BigDataCongress.2019.00014.

[152] Y. Hou, S. Garg, L. Hui, D. N. K. Jayakody, R. Jin, and M. S. Hossain, "A data security enhanced access control mechanism in mobile edge computing," *IEEE Access*, vol. 8, pp. 136 119–136 130, 2020. DOI: 10.1109/ACCESS.2020.3011477.

[153] H. Xiong, Y. Zhao, L. Peng, H. Zhang, and K.-H. Yeh, "Partially policy-hidden attribute-based broadcast encryption with secure delegation in edge computing," *Future Generation Computer Systems*, vol. 97, pp. 453–461, 2019, ISSN: 0167-739X. DOI: 10.1016/j.future.2019.03.008.

[154] Y. Pu, C. Hu, S. Deng, and A. Alrawais, "R²peds: A recoverable and revocable privacy-preserving edge data sharing scheme," *IEEE Internet of Things Journal*, vol. 7, no. 9, pp. 8077–8089, 2020. DOI: 10.1109/JIOT.2020.2997389.

[155] S. Dougherty, R. Tourani, G. Panwar, R. Vishwanathan, S. Misra, and S. Srikanteswara, "Apecs: A distributed access control framework for pervasive edge computing services," in *Proceedings of the 2021 ACM SIGSAC Conference on Computer and Communications Security*, New York, NY, USA: Association for Computing Machinery, Nov. 2021, pp. 1405–1420. DOI: 10.1145/3460120.3484804.

[156] Q. Zhang, M.-e. Xiong, P. Li, J. Yuan, and H. Zhu, "A secure data sharing model against keyword guessing attacks in edge-cloud collaboration scenarios," Mar. 2024. DOI: 10.21203/rs.3.rs-4011866/v1.

[157] U. S. Varri, D. Mallick, A. K. Das, M. S. Hossain, Y. Park, and J. J. Rodrigues, "Tl-abks: Traceable and lightweight attribute-based keyword search in edge–cloud assisted iot environment," *Alexandria Engineering Journal*, vol. 107, pp. 757–769, 2024, ISSN: 1110-0168. DOI: 10.1016/j.aej.2024.09.030.

[158] N. Wang, W. Zhou, Q. Han, J. Liu, W. Liao, and J. Fu, "A lightweight privacy-preserving ciphertext retrieval scheme based on edge computing," *IEEE Transactions on Cloud Computing*, no. 01, pp. 1–18, Sep. 5555, ISSN: 2168-7161. DOI: 10.1109/TCC.2024.3461732.

[159] Y. Zhu, X. Wu, and Z. Hu, "Fine grained access control based on smart contract for edge computing," *Electronics*, vol. 11, no. 1, 2022, ISSN: 2079-9292. DOI: 10.3390/electronics11010167.

[160] E. De Cristofaro, P. Gasti, and G. Tsudik, "Fast and private computation of cardinality of set intersection and union," in *Cryptology and Network Security*, J. Pieprzyk, A.-R. Sadeghi, and M. Manulis, Eds., Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 218–231. DOI: 10.1007/978-3-642-35404-5_17.

[161] G. V. Research, "Edge computing market size, share & trends analysis report by component, by application, by industry vertical, by organization size, by region, and segment forecasts, 2024 - 2030," Grand View Research, Tech. Rep. [Online]. Available: https://www.grandviewresearch.com/industry-analysis/edge-computing-market, (historical range: 2018-2023, accessed: 2024-10-30).

[162] A. Kayes, R. Kalaria, I. H. Sarker, *et al.*, "A survey of context-aware access control mechanisms for cloud and fog networks: Taxonomy and open research issues," *Sensors*, vol. 20, no. 9, p. 2464, 2020.

[163] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Communications Surveys Tutorials*, vol. 19, no. 4, pp. 2322–2358, 2017. DOI: 10.1109/COMST.2017.2745201.

[164] S. Lv, J. Ye, S. Yin, *et al.*, "Unbalanced private set intersection cardinality protocol with low communication cost," *Future Gener. Comput. Syst.*, vol. 102, pp. 1054–1061, 2020. DOI: 10.1016/j.future.2019.09.022.

[165] B. Carminati, P. Colombo, E. Ferrari, and G. Sagirlar, "Enhancing user control on personal data usage in internet of things ecosystems," in *2016 IEEE International Conference on Services Computing (SCC)*, San Francisco, CA, USA: IEEE Computer Society, 2016, pp. 291–298.

[166] R. Bohannon and A. Andrews, "Normal walking speed: A descriptive meta-analysis," *Physiotherapy*, vol. 97, pp. 182–9, Sep. 2011. DOI: 10.1016/j.physio.2010.12.004.

[167] S. Wilson, F. Schaub, A. A. Dara, *et al.*, "The creation and analysis of a website privacy policy corpus," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2016, pp. 1330–1340.

[168] Z. Wu and M. Palmer, "Verbs semantics and lexical selection," in *Proceedings of the 32nd Annual Meeting on Association for Computational Linguistics*, ser. ACL '94, Las Cruces, New Mexico: Association for Computational Linguistics, 1994, pp. 133–138. DOI: 10.3115/981732.981751.

[169] S. L. Miller and D. Childers, "Chapter 9 - markov processes," in *Probability and Random Processes*, S. L. Miller and D. Childers, Eds., Second Edition, Boston: Academic Press, 2012, pp. 383–428, ISBN: 978-0-12-386981-4. DOI: https://doi.org/10.1016/B978-0-12-386981-4.50012-6.

[170] Y.-S. Dai, B. Yang, J. Dongarra, and G. Zhang, "Cloud service reliability: Modeling and analysis," in *15th IEEE Pacific Rim International Symposium on Dependable Computing*, Citeseer, 2009, pp. 1–17. [Online]. Available: https://api.semanticscholar.org/CorpusID:11978736.

[171] K. Deb and H. Jain, "An evolutionary many-objective optimization algorithm using reference-point-based nondominated sorting approach, part i: Solving problems with box constraints," *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 4, pp. 577–601, 2014. DOI: 10.1109/TEVC.2013.2281535.

[172] W. Song, C. Zhang, and H.-A. Jacobsen, "An empirical study on data flow bugs in business processes," *IEEE Transactions on Cloud Computing*, vol. 9, no. 1, pp. 88–101, 2021. DOI: 10.1109/TCC.2018.2844247.

[173] K. B. Sheehan, "Toward a typology of internet users and online privacy concerns," *The information society*, vol. 18, no. 1, pp. 21–32, 2002. DOI: 10.1080/01972240252818207.

[174] O. Fennelly, C. Blake, O. FitzGerald, *et al.*, "Advanced musculoskeletal physiotherapy practice: The patient journey and experience," *Musculoskeletal Science and Practice*, vol. 45, p. 102 077, 2020, ISSN: 2468-7812. DOI: 10.1016/j.msksp.2019.102077.

[175] M. Hod, A. Kapur, D. A. Sacks, *et al.*, "The international federation of gynecology and obstetrics (figo) initiative on gestational diabetes mellitus: A pragmatic guide for diagnosis, management, and care," *International Journal of Gynecology & Obstetrics*, vol. 131, no. S3, S173–S211, 2015. DOI: 10.1016/S0020-7292(15)30033-3.

[176] B. Kacsmar, B. Khurram, N. Lukas, *et al.*, "Differentially private two-party set operations," in *2020 IEEE European Symposium on Security and Privacy (EuroS P)*, Los Alamitos, CA, USA: IEEE Computer Society, 2020, pp. 390–404. DOI: 10.1109/EuroSP48549.2020.00032.