PRIVATEME:
MANAGING PRIVACY IN MULTIPLE APPLICATIONS AND DEVICES

CHRISTIANNE HUBER

A THESIS SUBMITTED TO
THE FACULTY OF GRADUATE STUDIES
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
MASTER OF ARTS

GRADUATE PROGRAM IN INFORMATION SYSTEMS AND TECHNOLOGY
YORK UNIVERSITY
TORONTO, ONTARIO

OCTOBER, 2020

Abstract

Applications that tailor information to the user rely on data being collected and communicated. This may lead to privacy concerns about personal data and how it can be used. Even when privacy controls are available, it is not always clear which settings control the data collection. Furthermore, with the volume of data that is being collected, it is not always obvious how the data is collected.

Other researchers have proposed solutions to assist the user to manage privacy. Yet there is a need for a solution that will support privacy management on different devices, for multiple applications and web services.

PrivateMe includes a privacy goal model to capture privacy goals, a generic taxonomy of privacy settings and permissions, and an ontology to store reusable knowledge about how the privacy settings and permissions interact. PrivateMe is evaluated with three use cases that show its applicability to managing privacy on multiple devices.

Acknowledgements

I would sincerely like to thank Professor Marin Litoiu for supervising my efforts towards completing this thesis. I learned that the most effective way to learn and to stretch my abilities, is to complete unfamiliar tasks. I would also like to thank Professor Luiz Marcio Cysneiros for accepting to be on my supervisory panel. A sincere thank you also to Dr. Joydeep Mukherjee. Your patience and constructive criticism on the many drafts of this document, and your suggestions to add diagrams, have helped to make this document more organized and much easier to read. I would like to thank Dr. Mark Shtern for helping me to refine my thoughts into research questions. Thank you also for suggesting that I read documents such as the GDPR and NIST 800 myself. I would like to thank all my colleagues in the lab. I learned so much about cloud computing, containers, streaming data, and machine learning during my time there.

Last but certainly not least, I would like to thank my dear husband, my children, and other family members. It has been a long journey and completing this milestone would have been an even bigger challenge, if it were not for your support and encouragement along the way.

PrivateMe: Managing Privacy in Multiple Applications and Devices, was created independently, without sponsorship, affiliation, endorsement, or approval of Apple Inc. or Microsoft Corporation [1]–[3]. Google, Android, Google Play and Chrome are trademarks of Google LLC [4]–[6]. iPhone is a trademark of Apple Inc. [1]. iOS is a trademark or registered trademark of Cisco in the U.S. and other countries and is used under license [7]. All other trademarks are the property of their respective owners [3].

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

This research is motivated by the privacy concerns that arise from the volume of data that is being collected and used by applications. Preserving privacy is important as it protects fundamental democratic freedoms and rights. Managing privacy, however, can be complicated as privacy controls for various devices and applications can differ. Also, privacy is abstract and subjective, as illustrated in Figure 1. Additionally, there are other aspects of privacy that make it complex, such as time, context, and balancing one's privacy with the needs of society; these aspects are described in more detail in Section 2.1.

Figure 1: Privacy is subjective

1.1. Motivation

In this Section, I begin with examples of applications that use private data and the benefits that the applications can offer. Then I consider the compromises between these benefits and privacy.

I show that managing one's private data across multiple devices and applications can become complex. I discuss the privacy concerns and the issues that can arise from using applications, which lead to the research objectives in Section 1.2. I conclude this section with a brief introduction of PrivateMe.

I refer to the European Union (EU) General Data Protection Regulation (GDPR) as an example of legislation that is in place to protect private data. So, throughout this proposal, I use the GDPR term *data subject* to indicate the person about whom personal data is being captured, processed, or stored [8]. Although if data about a *user* is captured, the *user* becomes a *data subject*, the reverse is not necessarily the case, that is, the *data subject* may or may not be the *user*. I also use the GDPR terms: *data controller,* to refer to the person or organization that is collecting the data; *data processor,* to refer to the person or organization processing the data [8]. When referring to either a *data controller* or a *data processor,* or both, I use the term *service provider*. I discuss the GDPR more fully in Section 2.2.

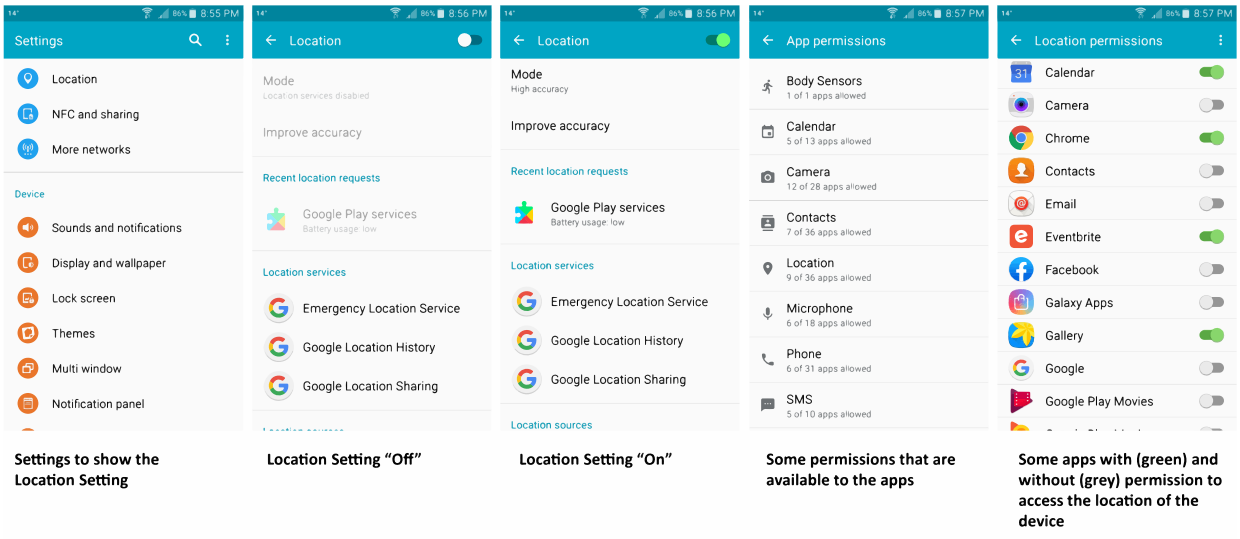


Figure 2: Screenshots of Privacy Settings and Application Permissions on author's mobile phone with Android™ [1]

Google and the Google logo are registered trademarks of Google LLC, used with permission [4], [10]–[13].

[1] The term "mobile phone with Android" is used as in the guidelines in [9].

Mobile devices, mobile applications (apps), and web services constantly gather, transmit, and process data to provide services that offer functionality and added convenience. As illustrated in Figure 2, the device settings control access to data such as the location, whilst the application permissions determine whether the application may access the data. For example, a financial services application requests access to the location of a mobile device to be able to find offices in the vicinity. If the *Location* device setting is "On" and the permission is granted to the application, whenever the user searches for nearby offices, the application simply accesses the location of the device and it shows the results on a map. If, however, the application does not have the required permission, or the *Location* device setting is "Off", the application requests that the permission be granted and that the *Location* device setting be switched "On", otherwise it requires user input. Another example is a photo repository web service that offers the convenience of automatically backing up the photos, with the added benefit of identifying and filtering photos based on the contents of the photos[2].

There are also several examples of the compromise between one's privacy and the benefits that one may realize when using applications. In the example of the financial services application, one must either switch the *Location* device setting "On" or input location information, thus giving up some location privacy. Once the *Location* device setting is "On", it may also enable the location to be accessed by other applications that request this permission. In the example of the photo repository web service, the potential threat to privacy is that over time, the photos could provide information about the time of year one goes on vacation and whether one has a favourite vacation spot. One may select the privacy option for the web service not to store location history and not to track activity. Nevertheless, photos that contain a distinctive landmark in the background, which

[2] https://www.google.com/photos/about/

could be identified by the photo web service functionality [14], could provide details of one's location history and one's daily activities, particularly since the date of the picture is also stored. If facial features are analyzed to determine which photos contain a particular data subject, this data could be captured by the data controller too.

Attempts at managing one's private data can become rather complex as it may lead to conflicts between the privacy settings on different devices or between permissions for multiple applications. When a data subject uses multiple devices, applications, and web services, the privacy setting or permissions selected for one of these may affect the selections previously made for another. For example, if the financial services application and a shopping application both start with the *Location* permission granted, while the *Location* device setting is "Off", then neither application has access to the location of the device. If the *Location* device setting is switched "On", the financial services application is permitted to access the device location but so is the shopping application. The data subject, however, may be unaware that the shopping application now has this permission. If the data subject does not want to permit the shopping application to access the location, this causes a conflict between the data subject's privacy goals and the *Location* permission of the shopping application.

Although data collection is necessary for applications to function, it can lead to privacy concerns such as indirect data collection, inference, aggregation, and unethical use of the data. According to Sengul [15], indirect data collection occurs when data is being captured for a particular purpose but it also inadvertently collects additional data. Vojković [16] provides the example of video surveillance in a smart city indirectly providing the means to identify data subjects by the cars that they drive. If the video is being recorded to model traffic patterns, it is not necessary to identify the drivers; the data should be restricted to only what is necessary by

pseudonymizing, or preferably by anonymizing, the personally identifiable data [16]. Inference occurs when the available data is used to ascertain additional pieces of information [17]. For example, Fernquist et al. [18] shows that metadata such as time and event profiles can be used to infer the identities of IoT users, even though the metadata does not include data about the user[3]. Observing the data over a longer period of time, increasing the volume of data, and aggregation, that is, combining different pieces of data, also improves the accuracy with which the user can be identified [18]. Finally, it is considered "unethical" to disseminate the data [19] or to have a secondary use for the data, that is, to use it for purposes other than that for which it was originally collected, without obtaining explicit consent from the data subject [20], [21].

Data protection and privacy laws such as the European Union (EU) General Data Protection Regulation (GDPR), aim to protect the privacy of data subjects in the collection, processing, use and storage of personal data [8]. The GDPR applies to all EU companies and to any international companies that conduct business in the EU, that process personal data of EU subjects [8], [20]–[22].

Despite legislations such as the GDPR, data subjects may still remain unaware of how, why, and when the data is collected, or how it will be used in the future [18], [23], [24]. According to Anthonysamy et al. [25], there are some discrepancies between the privacy policies of some social network sites and the privacy permission controls that are offered to the data subjects. Zimmeck et al. [26] finds that there is a difference between the data collected and used by some mobile applications, and what is stated in the privacy policies. Additionally, Almuhimedi et al. [27] indicates that data subjects change their app privacy permissions once they are aware that data is

[3] Note that here I use the term *user* rather than *data subject* because in this case, the data does not include data about the person.

being collected. This makes it necessary for the data subject to trust that the service provider[4] will only use the data for the intended purpose [21], [28]. Accordingly, when the service provider is accountable for the use and protection of the data, it helps the data subject to trust the service provider and the technology [19], [21].

For a data subject to manage and control his or her own data, Colley and Crabtree [19] offers a possible solution, and Frecè [28] surveys currently available solutions, however these are for only one particular type of application. Self-adaptive systems for privacy have also been proposed in [23], [24], [29], however, these rely on the application itself to manage private data. The solution in Almuhimedi et al. [27] monitors and warns the data subject when data is collected by apps on a device. It is, however, limited to the one device. The solutions offered by [30], [31], are designed for ubiquitous computing environments. Langheinrich [30] has proposed using a private proxy to manage data dissemination in a ubiquitous computing environment, when data is requested by other proxies. Schaub et al. [31] proposes a solution for autonomous privacy management in ubiquitous computing, which adapts to changes in context and makes recommendations to the user when needed. Although [31] helps the data subject to manage multiple devices, including mobile phones, it does not describe how any conflicts between the device settings and app permissions are resolved.

The related work discussed above protects privacy by limiting data collection and sharing. While Tun et al. [29] associates the private data with functionality, [23], [24] balance privacy with benefit to the data subject, and [31] discusses trusted entities in the environment. To the best of my knowledge, none of the related work uses all of these and other data subject privacy goals, to determine how to manage the private data. Nor do they address the interaction of the device

[4] Here, *service provider* indicates the data controller or the data processor or both.

privacy settings with the privacy permissions of applications or web services. I propose PrivateMe as a technique to assist a data subject with managing privacy settings on multiple devices, and permissions for multiple applications and web services.

1.2. Research Objectives

The concern about how much data is being collected, how it is being collected, how it will be used, and by whom, has led to the following research questions:

RQ1: How can a data subject ensure that the selected privacy settings and permissions correctly reflect the data subject's privacy goals?

RQ2: How does a data subject determine if there are any conflicts between the privacy settings and permissions?

RQ3: How can a data subject manage various privacy settings and permissions on multiple devices, applications, and web services?

1.3. Research Contributions

With this research I address the difficulty that a data subject may have with managing privacy settings on multiple devices, in multiple applications, and for multiple web services. PrivateMe is a technique that includes a softgoal interdependency graph (SIG), a taxonomy, and an ontology. SIGs are used to illustrate and represent non-functional requirements (NFRs) as softgoals, to show how the NFRs affect each other (their interdependencies). The PrivateMe SIGs capture the privacy softgoals. The data subject uses the SIG to assess the privacy softgoals and their interdependencies, to decide on his or her privacy requirements. The data subject refers to the generic taxonomy of privacy settings to determine which settings to use to manage privacy on the

data subject's device, and for the application or web service. By including knowledge about how the privacy settings interact, the resulting ontology can be used to ensure that the privacy settings and permissions support the data subject's privacy goals. Furthermore, it can be used to identify any conflicts between the settings and permissions.

The contributions of this thesis are as follows:

- A softgoal interdependency graph (SIG) that captures the privacy goals
- A taxonomy of privacy settings, common among multiple applications and devices
- An ontology that captures the knowledge of how the settings and permissions interact

PrivateMe is evaluated with three use cases. In Use Case 1, the data subject refers to PrivateMe to ensure that her privacy goals are met as she decides on the appropriate settings and permissions to select for an application on her mobile phone. In Use case 2, she does not want the settings and permissions from Use Case 1 to conflict with the settings and permissions for another application on the same device. In Use Case 3, the data subject wants the settings and permissions for the application and the website, to be consistent on all her devices. She uses PrivateMe to help her decide which settings and permissions will support her privacy goals, to help her find the settings and permissions on her devices, and to identify any conflicts between them.

1.4. Thesis Organization

I have discussed the concerns about privacy, resulting from how much data is being collected by applications, as the motivation for this research leading to the Research Objectives. I have described the Research Contributions. In the remainder of this thesis, Chapter 2 contains a discussion of the Background and the Related Work. PrivateMe is presented in Chapter 3. Chapter 4 describes three use cases that evaluate PrivateMe. Chapter 5 concludes this thesis.

Chapter 2: Background and Related Work

Privacy is a word that is used in everyday language, but one's sense of privacy is rather subjective. To understand *privacy*, in Section 2.1 I first define the term and then discuss aspects of *privacy* that make it complex. I provide examples of private and sensitive data that are protected by legislation and I consider why privacy is necessary. Since I refer to the GDPR throughout this document, I provide some further details of the GDPR requirements in Section 2.2. Section 2.3, ends this chapter with a discussion of the Related Work.

2.1. Privacy

The Cambridge Dictionary defines *privacy* as "someone's right to keep their personal matters and relationships secret" or "the state of being alone"[32]. Similarly, Merriam-Webster dictionary defines *privacy* as "the quality or state of being apart from company or observation", "seclusion" or "freedom from unauthorized intrusion"[33]. Solove [34], however, argues that if one is secluded, privacy is not an issue, since *privacy* is about the person in a community, and freedom from interference into private matters, such as one's home or one's person. The concept of privacy is abstract and subjective. As illustrated in Figure 1, different individuals have different preferences as to how much personal information they share.

The term "private sphere" is used to suggest personal surroundings, within which one's private information is disclosed. Similarly, in Figure 3, I illustrate information about the data subject as a sphere. Different views of private and public information are illustrated as spot lights on particular portions of the sphere. The data subject has a complete view of the data. Certain data controllers have a view to some private data, whilst others have a view of some sensitive data,

which may also be accessed by data processors, if and when necessary. The social view and the public view may overlap, however the public view and private view do not.

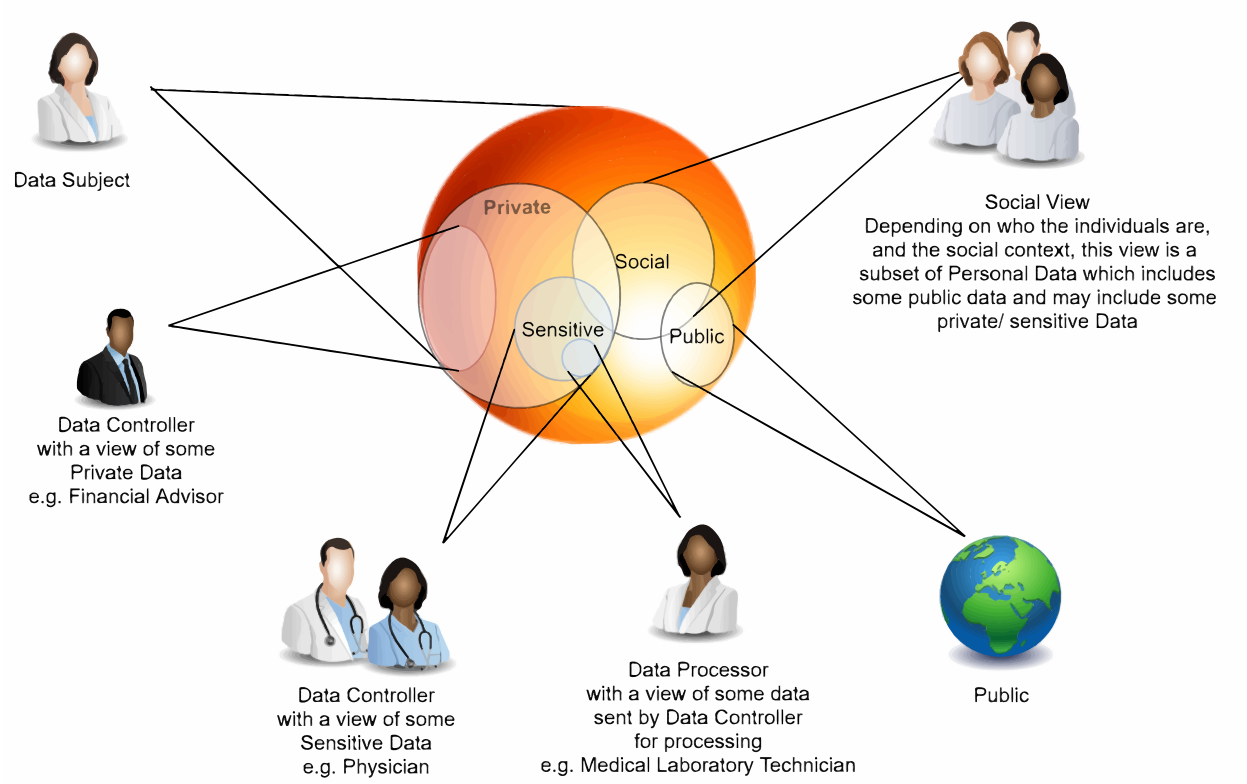


Figure 3: Different Views of One's Personal Data

Privacy needs change depending on the environmental context. The context includes what the individual is doing, where the user is, and whoever or whatever is in the vicinity [29], [31]. For example, one may be willing to share details of a private issue at home or with someone who has been a close friend for several years. That same individual, however, may be unwilling to share such details at work, with a colleague or an acquaintance.

Privacy changes over time. It is influenced by culture and as it has become more commonplace to have space available for private activities, historical concepts of privacy have also changed [34]. There are, however, other temporal aspects to privacy which differ from the historical changes in

the concepts of privacy. Palen and Dourish [35], discuss managing privacy within boundaries of *Disclosure*, *Identity*, and *Temporality*, to control what private data is disclosed, by and to whom, and when. To accommodate changes to privacy goals due to changes in context and time, [29] uses privacy norms to specify conditions, such as where, when, and to whom, the data subject's information may or may not be disclosed or inferred from other data. As indicated in [35], the use of technology changes how data is captured and stored; it makes the information more readily available, possibly to a wider audience than it would have been otherwise.

In analyzing theories of privacy, [34] indicates that there is a complexity to *privacy* that leads to aspects of privacy being omitted if theories are too narrow; conversely, if the theories are too broad, they fail to provide a precise enough definition of *privacy* to be used in privacy protection laws. Legislation protects personal information, such as age, financial information, and location, as well as sensitive information, such as culture, health, religion, and political beliefs [8]. Privacy in such matters is necessary to protect and maintain one's dignity, one's fundamental democratic freedoms, and one's personal safety [8], [18], [34]. According to [34], some law courts have maintained that once something is public, it is no longer private. The GDPR, however, has a stricter definition of what personal data is public data; it is public as long as the person him or herself makes it public [8].

A data subject has the right to control who has access to the personal data, by consenting to the collection or use of the data [22]. Nevertheless, the data subject's right must also be balanced by the rights of others and of society [8], [20], [21], [34]. For example, although a person may not wish to share health information with others, if the person is a patient, it is necessary to disclose pertinent information to the attending physician.

Privacy is complex. It depends on context, time, and a balance between the needs of the individual and the rights of others. Privacy legislation requires that private and sensitive data is protected to ensure that a person's rights and freedoms are maintained.

2.2.GDPR

We refer to current legislation such as the GDPR, which became effective May 25, 2018 [16], to obtain information about the data subject's rights and the responsibilities of the data controllers. As it is an EU regulation, the GDPR applies to all companies established in the EU that process personal data of EU subjects [8], [20]. However, since the data could be stored and/or processed internationally, according to its "Territorial scope", the GDPR also protects these rights with international companies that do business in the EU [8], [20]–[22]. It is applicable regardless of whether any payment is made for the services [8].

As part of the data protection, the GDPR requires data controllers to be accountable for ensuring that the GDPR principles are met, and to have the appropriate privacy policies in place to protect personal data [8]. The data controller and data processors are also responsible for ensuring that data protection is maintained by design, and that protection for the data is the default [8], [20], [21], [36]. The GDPR includes six principles for which the data controller is responsible: Lawful, fair use and Transparency, Purpose Limitation, Data Minimization, Data Accuracy, Storage Limitation, and Integrity and Confidentiality [8].

To maintain transparency, the data controller must provide the terms of data collection, including the type of personal data that is being collected and processed, and the reason for collecting the data [8], [20]–[22]. Limiting the use and retention to the specified purpose, contributes towards the principle of Purpose limitation [16], [21]. Similarly, collecting only the

personal data that is necessary, and retaining the personally identifiable data for only as long as it is required, contributes towards upholding the principles of Data minimization and Storage limitation [16], [21]. Removing personal data, for example by applying data pseudonymization, should be done as soon as possible [8], [16], [21].

The GDPR protects the data subject's rights by specifying the conditions necessary for the average person, as a data subject, to make an informed decision when consenting to the use of personal data. The data controller must provide the terms of data collection in easy to understand language [8], [16], [22]. The request for consent must be made in clear, simple language; it should be easy to access, and separate from any other information [8], [22]. The data controller must request additional consent for any new additional use of the data [20], [21]. When providing consent, the data subject, must do so freely and explicitly and must also be able to easily withdraw the consent [8], [21], [22]. If machine decision-making or profiling is used, the data subject may have the right to object and to request that the decision is made by a natural person [8], [22]. The data subject's rights, however, need to be balanced with the rights of others and society, and the legal responsibilities of the data controller [8], [21], [22].

Although the GDPR principles of Lawful, fair use and transparency, the Purpose limitation and the Data minimization principles may be according to the data controller's terms of data collection and use, there remains a complexity to managing one's privacy goals; it is possible that a person's intended privacy goals are still not met.

2.3. Related Work

In this section, I discuss related work that use models of *Privacy*. Since privacy policies are an important way that service providers communicate to the data subjects about how private data

is protected, I discuss related work that assesses these privacy policies. I then discuss some solutions that allow the data subject to manage his or her own data privacy as well as some solutions that autonomously manage privacy goals at run-time. Throughout this section, I discuss the models that are used, the privacy goals, threats, and settings or permissions available to the data subject to manage privacy.

2.3.1. Privacy Models

Other researchers use models to illustrate privacy requirements and typical solutions, to build business processes that support the privacy requirements, and to ensure that the privacy requirements are met. Here, I briefly describe some of this related work and I discuss how these methods may be applied to aid a data subject in a similar manner.

The Cambridge Dictionary defines "threat" as "the possibility that something unwanted will happen, or a person or thing that is likely to cause something unwanted to happen" [37]. Solove [34] reviews various legal cases to classify different aspects of *Privacy*, by using a "pragmatic", "bottom-up" method. By reviewing legal cases, the courts' assessment of the harm done, and the financial, emotional, and social impact of the harm to the privacy victim/s, his resulting "Taxonomy of Privacy" is one of privacy *threats.* I describe how I extend on Solove's Taxonomy of Privacy (threats) in Section 3.1.

Veleda and Cysneiros [38] present a tool to help requirements engineers to elicit non-functional requirements (NFRs), and to help software engineers search the existing knowledge for typical solutions to the NFRs, and how they affect each other. In [38], *Privacy* is one NFR along with other NFRs including *Security* and *Trust*. Zinovatna and Cysneiros [39] contribute the knowledge of *Privacy* and *Transparency*. Both [38], [39] indicate that softgoal interdependency

graphs (SIGs) can become large. Nevertheless, [38], [39] illustrate the NFRs as SIGs. According to [38], it is easier to view the graphical models than to find the knowledge in text-based references.

Kalloniatis et al. [40] introduce a methodology to include privacy requirements in the system design and to match the privacy solutions to the requirements. The goal model is arranged as a hierarchy of strategic goals which are broken down into operational goals and business goals [40]. Their work identifies a set of eight privacy and security goals (to protect the private data) and the corresponding processes that are typically used to maintain the required privacy [40]. The business processes are adjusted to include the privacy processes and the appropriate privacy technology solutions are selected [40]. Thus, [40] proposes a method in which the privacy solutions are selected during the design phase of a system to ensure that the solutions match the privacy goals.

Liu et al. [41] propose a framework that uses goal models and a methodology to capture and analyze security and privacy requirements. First, [41] identifies the actors, then models the requirements in i* (iStar) strategic rationale models (SR), and the dependencies of the actors (or roles) in strategic dependency models (SD). To conduct the security and privacy analysis [41] determines how access for each role could be abused. According to [41], the dependencies indicate where there may be a weakness that could be exploited. Then, [41] refers to existing knowledge to identify the corresponding threats and typical mitigation strategies. To select the appropriate solutions, [41] evaluates how the options that are available to mitigate the threats, affect the goals. Finally, [41] assigns role-based access control which they assess based on least privilege and separation of duties.

Ahmadian et al. [42] propose a methodology that uses models to perform a privacy impact analysis (PIA). The PIA is conducted during the design phase of a system. The analysis step is described more fully in Ahmadian et al. [36]: the system Unified Modeling Language (UML)

models are analyzed to identify the private data, where it is processed, and that it is protected according to the privacy level agreement (PLA). A PLA specifies the level of privacy protection a data controller negotiates with the data processors [36], [42]. The next step is to identify the security and privacy threats [42]. The impact is calculated based on the sensitivity of the data. To mitigate the risks, the appropriate security standards and controls are applied. Besides presenting a methodology for conducting a PIA, the work in [36], [42] shows that the use of models helps to identify privacy threats and that these models and threat assessments help increase privacy awareness.

The knowledge and the methods contributed by [38]–[41] include goal models to aid the software engineers during the design phase of a system, to analyze privacy NFRs and to select the appropriate solutions to meet the requirements. Although [36] includes a step that identifies private data, both [41], [42] identify threats to conduct a threat assessment. The PIA in [42] is also done during the design phase of a system. Once the system is operational, however, the data subject may be presented with privacy settings to manage privacy. Therefore, the data subject also needs to understand what data is managed by the options that are offered, as well as the possible privacy threats. Furthermore, the data subject needs to be able to assess the threats effectively to select the appropriate option.

The work in this thesis differs from [36], [38]–[42] since I use goal models to aid the data subject to assess the privacy options and to ensure that the selected options support the data subject's privacy goals. The SIGs capture the data subject's privacy goals. To elucidate the privacy threats, I model them separately from other goals. By including in the SIG, the type of data that make these threats possible, the data subject can understand the impact of releasing the

data. Thus, the data subject can assess the threats against other goals such as personalizing the application.

2.3.2. Privacy Managed by the Data Subject

Data controllers are required to be transparent about the collection and use of personal data and to limit it to what is necessary for the specified purpose [8], [20], [21]. This includes providing information about what data is collected, where it is processed, by whom, for what purpose [16], [20], [22]. Such information is typically provided in privacy policies [25], [26]. In this section, I describe the related work that examines privacy policies. Then, I discuss work that aims to enable the data subject to manage personal data. I conclude this section with a discussion about how PrivateMe differs from these solutions.

Makri and Lambrinoudakis [43] propose a privacy audit methodology that focuses on data controller transparency with documentation such as privacy policies. According to [43], conducting an audit and making the audit results and documents available to the data subject, promotes trust [43].

Zimmeck et al. [26] contribute an automated method to audit the privacy policies of mobile apps. The authors suggest that the audit can be used by authorities, as well as by data controllers, to check that they are transparent about their data collection and use [26]. According to the authors, although [26] is done on mobile apps for Android, it can be applied to apps for iOS and site cookies. The audit in [26] determines which apps have privacy policies. It classifies the apps according to whether the policies inform the data subject about changes, as well as whether the policies include information about how the data subject may control the data [26]. It also classifies the policies according to the terms for personal data collection and processing, and data sharing [26].

Additionally, [26] analyzes how the apps collect, use and disseminate personal data and ascertains that the majority of the apps that do not have a policy, should have one. The same analysis of the apps that do have a privacy policy, and subsequent comparison to what is stated in the privacy policy, shows that there are some variances between what is stated in the policies and what the apps do [26].

Anthonysamy et al. [25] studies the privacy policies of several social network sites; it matches what is stated in the privacy policies to the site settings that manage data privacy. The work in [25] finds that the settings are not always sufficient to mange the personal data, and that some policies may be ambiguous regarding how additional data is collected and shared with third parties.

Frecè [28] reviews various options for data storage. These range from personal data storage (with no data sharing), to options which allow the data subject to determine who has access, and finally to solutions which autonomously manage data sharing on behalf of the data subject. In doing so, [28] highlights the issue of trust and the opposing goals of the data subject and the data controller. As indicated in [28], the options for a data subject to consider when selecting data storage are interdependent. Therefore, options such as the business model, encryption, and access to the data for analytics [28], need to be balanced to achieve the data subject's privacy goals.

Colley and Crabtree [19], present Object Based Media and Databox. The system works with IoT devices in the environment to provide multi-modal media that adapts and is personalized to each person in the room [19]. To meet the GDPR requirement of accountability to the data subject, the Databox has a dashboard which provides the means to control personal data [19]. The data subject authorizes the data processing which is then done locally on the data subject's databox; only the results of the processing are transmitted to the data controller [19].

Almuhimedi et al. [27] presents a solution to alert the user (data subject) when private data is collected by apps on an Android 4.3 or 4.4 device. The study conducted by [27], shows that the data subject responds to the prompts by reviewing and revising the privacy permissions of the apps. The prompts help the data subject to know which apps are collecting data, how often data is collected, and what type of data is being collected [27].

For the data subject to control private data, first, the data must clearly correspond to the service that is being offered. Then, the data subject needs to understand and assess the possible threats. Finally, the data subject must have the privacy settings to manage the data. The audits and assessment of privacy policies in [25], [26], [43] promote data controller transparency. According to [25], [43], this helps the data subject to trust the data controller and the service. The audit in [43], however, does not associate the data use with possible threats. In [26], personal data is associated with the possible threats of data collection and dissemination. Since [26] audits apps, it considers the capability of the apps to collect and process personal data. An audit, however, does not check the run-time behaviour of a particular installation, with settings and permissions granted or denied by the data subject. Although not indicated as potential threats, [25] lists data collection and dissemination, as well as indirect data collection, aggregation, and personalization. According to [25], there is a need for privacy controls to match the corresponding data that is being collected and used. Neither [25], [26], however, assist the data subject to select the appropriate controls (or permissions), nor do they consider the interaction between device settings and the app or site permissions. Although [28] mentions threats to data storage options, such as access to decrypt the data and unauthorized use, it does not include the privacy settings of the or the privacy permissions of the app that is used to access the data. In [19], the Databox dashboard enables the data subject to control privacy. Still, [19] does not include a solution to help the data subject relate

the data to possible threats or to configure the privacy settings. To assess the threat resulting from releasing the data, the data subject needs to understand the functionality that the data collection supports. As noted in [27], three participants in the field study had to revise the permissions previously removed, since the functionality they needed was lost.

My work differs from [19], [25]–[28], [43] since I use SIGs to illustrate the privacy goals. The SIGs associate the data with the benefits from the service but also to the possible threats. The SIGs also model other privacy goals to be balanced against the possible threats, such as trust. This guides the data subject to consider the threats that may arise from permitting the use of the data. The data subject can then decide whether to trust the service. My work also focuses on how privacy settings are presented to a data subject: I provide a taxonomy of the privacy settings and permissions and I include an ontology to store the knowledge of how the settings and permissions interact. This helps the data subject to control the private data according to the privacy goals and to manage the interaction between the settings and permissions of the device and the app, as well as between multiple devices, apps, and sites.

2.3.3. Managing Changes to Privacy Goals due to Changes in Context

In this section, I discuss some related work that proposes systems that address changes to the data subject's privacy goals due to context changes. I discuss how the proposed solutions identify threats and assess them, then adapt to mitigate the threats. I conclude this section by indicating how PrivateMe differs.

Tun et al. [29] discusses the issue that the privacy requirements of a data subject depend on context and therefore change over time. The work of [29] contributes privacy arguments that an adaptive system could use to reason about the context. In doing so, [29] addresses the temporal

aspect of privacy. According to [29], the context includes the roles of the data subject and data recipient, the location, and time. These factors are used to capture the privacy norms (rules, or goals) of the data subject [29]. The privacy arguments describe the circumstances under which to apply the privacy norms [29]. They have parameters so that the conditions can be tailored by the data subject [29]. These arguments are implemented using event calculus so that the system can reason about the context to ensure that the privacy goals are met [29]. When the recipient requests data, the system ensures that the privacy norm is met in the current context and that other data cannot be used to reveal additional information that would negate the privacy norm [29]. Thus, [29] selectively discloses information to the recipient.

Omoronyia et al. [24] contribute a tool for software engineers to design systems that adapt to changing context. I discuss it here, since the privacy awareness requirements (PAR) described in [24] are implemented as part of the framework in [23].

Omoronyia et al. [24] suggests that the context needs to be monitored continually to identify privacy threats. Since it can be difficult for a data subject to control private data, [24] recommends a self-adaptive system that uses selective disclosure to manage personal data. According to [24], the attributes used to determine a change in context, may themselves, also change. The self-adaptive system first determines which attributes (PAR) need to be monitored, then it conducts a threat analysis, and finally completes a threat assessment [24]. To determine the PAR, [24] finds attributes that are specified in the privacy policies, as well as the domain model and the behavioural model. The threat analysis examines current context and data previously obtained by the recipient to identify threats that will not meet the privacy goals specified in the privacy policy [24]. The threats are then assessed to determine severity [24]. The system reasons about privacy based on

threat severity as it simulates the transitions in state; it makes recommendations to the system designer accordingly [24].

Omoronyia et al. [23] describe their framework for self-adaptive systems to address privacy requirements. As in [29] they use selective disclosure as a solution. Their approach, however is one in which privacy is offset by the functional benefit to the user [23]. The context model describes the domain, that is, the entities, their relationships, and their attributes; the behavioural model describes their current state and how they transition from one state to another [23], [24]. They identify the PAR by finding the subset of attributes that are in the context model, the behavioural model, and the data subject's privacy goals [23]. They identify the privacy threat based on the context and the history of information previously sent between the data subject and the recipient [23]. They assess the severity of the privacy threat based on the social connections of the recipient, and evaluate it against the benefit to the data subject if the information were to be disclosed [23].

Schaub et al. [31] acknowledge that solutions for protecting privacy are difficult to configure, and that changes in context make it difficult to define privacy goals in advance. They highlight the issue of data collection in a ubiquitous computing environment being unobtrusive to the data subject and therefore, "passive" [31]. The solution proposed in [31] provides a way for a data subject to manage multiple devices in the environment. The context model includes the data subject, resources (sensors or devices), activities of the data subject and the entities with whom or which the data subject interacts, their states, and the environment [31]. The privacy decision engine is initiated based on personality type [31]. It evaluates a change in context by comparing it to cases stored in the knowledge base [31]. It infers the privacy goals based on trusted entities within the context and privacy policies which state multiple ways to meet privacy goals of the user

[31]. A confidence score threshold is determined based on previous decisions as well as the data subject's personality and response to the decisions [31]. It either autonomously selects an option, or if the confidence level is below a threshold, it requests user input [31].

In [23], [24], [29], the focus is on one application that adapts to the context to manage data subject privacy. They associate the data with the functionality that the app offers, and mitigate the threat of data disclosure to other users of that application. [23], [24] also conduct a threat analysis and assessment. None of these solutions, however, help to identify and mitigate any other types of privacy threat or other aspects relating to privacy, such as trust. Even though [29] mentions multiple apps as part of their future work, [23], [24], [29] do not address the interaction between the device privacy settings and privacy permissions of multiple apps and sites. Whereas [31] manages data for multiple devices, it does not specify whether a data subject is made aware of the corresponding potential privacy threats, or how the data can be used to benefit the data subject.

In PrivateMe, I use SIGs to capture privacy goals as well as the potential privacy threats. The SIGs associate the data with the functionality that is offered by the data controller and that could be of benefit to the data subject. The SIGs also associate the data with the potential privacy threats. Thus, by illustrating the benefit that can be offset against the possible threat, the SIGs can guide the data subject to decide whether to permit the use of the data. The taxonomy helps the data subject to find the settings and permissions that manage the data. The knowledge captured in the ontology, of how these settings and permissions interact, helps the data subject to ensure that the selected settings and permissions do not result in conflicts between multiple devices, apps, and sites.

Chapter 3: PrivateMe

I propose PrivateMe to provide a technique that can be used to manage privacy settings on multiple devices, and permissions for multiple applications and web services. It includes a softgoal interdependency graph, a taxonomy, and an ontology. The softgoal interdependency graph represents the privacy goals; it shows what data is used to realize the softgoals, and how the data contributes towards the privacy softgoals. The taxonomy provides a generic classification of device privacy settings and permissions. The ontology captures the knowledge of how the privacy settings and permissions interact. This chapter is organized as follows: In Sections 3.1, 3.2, and 3.3, I describe the main components of PrivateMe: the softgoal interdependency graph, the taxonomy, and the ontology respectively. In Chapter 4, I show how I evaluate PrivateMe with three use cases.

3.1. Softgoal Interdependency Graph

To ensure that the privacy settings and permissions selected by a data subject correctly reflect the data subject's privacy goals (RQ1), it is necessary to compare the settings and permissions with the privacy goals. First, however, one must know what those privacy goals are, as well as what data contributes towards them. The privacy goals must be represented in an effective manner such that one can assess and balance them with other privacy concerns. In this section, I briefly discuss the characteristics of non-functional requirements (NFRs). Then, by reviewing the characteristics of privacy goals and comparing them to NFRs, I explain why privacy goals are NFRs. I discuss the use of Softgoal Interdependency Graphs (SIGs) and the NFR Framework [44] to represent NFRs as softgoals. I list and organize the concerns that are mentioned in the literature to illustrate them in the SIGs that are used with PrivateMe. I provide an example of a

PrivateMe SIG. I conclude this section by explaining how the SIGs address the research objectives.

As discussed in Chapter 2, privacy is abstract, subjective, depends on context, and it can change over time. Privacy legislation such as the GDPR applies to all software. Furthermore, within any one particular piece of software, privacy goals apply regardless of the functionality that is being used. The previous chapters provided examples of compromises between maintaining privacy and the benefits that can be gained by using some private data. So, privacy needs must be balanced with other concerns. Since privacy is subjective, each data subject will have different privacy goals. Therefore, privacy is complex and privacy goals are difficult to articulate.

Non-functional requirements (NFRs) are often referred to as quality requirements or constraints that apply to the system regardless of what functionality is being used [45]–[47]. These constraints include regulations [46], [47] such as privacy legislation [47]. NFRs can also affect each other, making it necessary to compromise between them [44], [45]. According to Chung et al. [44], it is typical for NFRs to be subjective but this makes it difficult to measure how well the NFRs are satisfied. Since privacy goals apply regardless of what functionality is being used, they are subjective, and they may need to be balanced with other concerns, privacy goals are, therefore, also non-functional requirements.

The NFR framework [44], is used to represent NFRs in Softgoal Interdependency Graphs (SIGs), so that they can be analyzed and appropriate solutions can be selected. Since NFRs are subjective, we refer to them as softgoals [44]. Due to compromising between softgoals, we accept them being sufficiently satisfied or *satisficed* [44] . I use the NFR Framework to create the SIGs in PrivateMe because the SIGs provide a visual representation of softgoals. With the NFR Framework, the softgoals are broken down systematically into more specific softgoals and

alternative methods of satisficing those softgoals are included, so that can then be assessed and decisions between the alternatives can be made. In the following paragraphs, I summarize parts of the NFR framework, as described in [44], which I use in PrivateMe.

High-level NFR softgoals can be broken down in a systematic manner, or *decomposed,* into a hierarchy of more specific softgoals (subgoals). An *AND decomposition link* between a softgoal and its subgoals, indicates that all subgoals must be satisficed for the softgoal to be satisficed. An *OR decomposition link* indicates that at least one subgoal must be satisficed for the softgoal to be satisficed. Decomposition of the softgoal in this manner, *explicitly* states how the subgoals contribute towards the higher-level softgoal. *Explicit interdependencies* such as these are indicated in the SIG as solid lines [44].

Possible solutions for a softgoal are illustrated in the SIGs as *operationalizations*. These operationalizations can be processes, functions, or even data that is used to satisfice the softgoal [44]. The operationalizations may be decomposed in a similar manner to the softgoals. The *explicit interdependencies* between the softgoals and the operationalizations which are added to the SIG as solutions to the softgoals, are indicated with a solid arrow; the solid line shows that it is an *explicit interdependency* and the arrow indicates the direction of the interdependency. The following *contributions* indicate how the operationalizations affect the softgoal: strongly negative (Break or --), negative (Hurt or -), neutral or unknown (?), positive (Help or +), or strongly positive (Make or ++). The S- is used to indicate Some Negative and can be used to show Break or Hurt. Similarly, the S+ indicates Some Positive and can be used to show Help or Make.

Softgoals, however, also affect each other. These *correlations* between different softgoals, between operationalizations and other softgoals, or between different operationalizations, also need to be assessed. According to [44], the correlations are *inferred* based on existing knowledge,

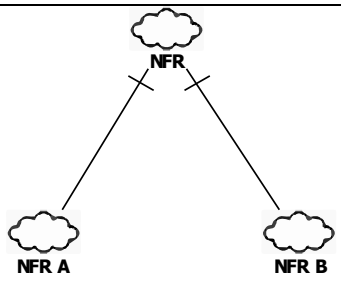
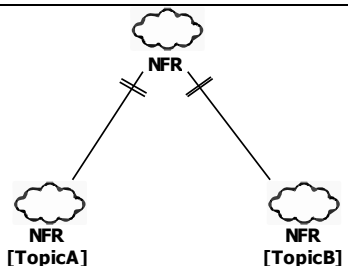
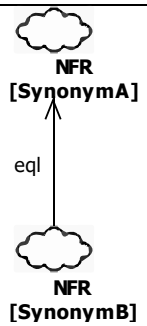
experience, expertise, or some combination thereof. As such, they are documented in the SIG as *implicit interdependencies*, with dotted lines [44]. The contribution of the correlation can be indicated in a similar manner to the explicit interdependencies [44]. For example, if a softgoal, A, works positively towards another softgoal, B, and softgoal A is satisfied, then softgoal B is *satisficable*; if softgoal A, works against softgoal B, and softgoal A is satisficed, then softgoal B is *deniable* [44].

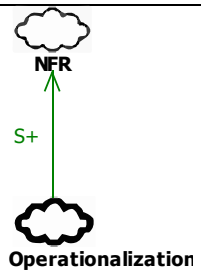
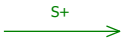
When designing a software system, decisions are made to select the operationalizations that will be used. The reasons for these decisions are illustrated in the SIGs as *claims*. The design decisions, *targets,* are then translated into functional requirements [44], [48].

RE-Tools [49] is open source modelling software which is used for requirements engineering. It is available as a plugin to the open source software version of StarUML [50]. I use RE-Tools to apply the NFR framework and create the SIGs for PrivateMe. The softgoals, their interdependencies and correlations, with their contributions, are illustrated in the SIGs according to the legend in Table 1. The SIGs in PrivateMe are only intended to help visualize the softgoals and the interdependencies; they are not used to obtain functional requirements (FRs) for a new system. As such, in Table 1 below, I do not include all the icons used in the NFR framework.

Table 1: Softgoal Interdependency Graph Legend *[44]* using the icons available in *[49]*

Icon	Meaning
	NFR Softgoal

Icon	Meaning
	Operationalization Softgoal
	Claim Softgoal
	Contribution Link: AND decomposition Decomposition may be on NFR type or NFR topic, here illustrating decomposition on NFR type.
	Contribution Link: OR decomposition Decomposition may be on NFR type or NFR topic, here illustrating decomposition on topic. Further decomposition on topic may also be done and illustrated as [Topic.Attribute].
	Contribution Link: Equal

Icon	Meaning
	Contribution May be any one of the following: S- Some – (range includes Break and Hurt) - - Break – Hurt ? Unknown S+ Some + (range includes Help and Make) + Help + + Make
	Explicit Interdependency (Contribution may be as above)
	Implicit Interdependency (correlation) (Contribution may be as above)

In the following paragraphs, I describe the method to determine the concerns included as softgoals in the SIGs for PrivateMe. Then, I use the Trust SIG as an example to describe the use of the NFR Framework in developing the PrivateMe SIGs.

To determine which concerns to include in the PrivateMe SIGs, I gather the privacy concerns discussed in the literature and tabulate them as in Appendix A. The concerns include operationalizations that help mitigate the privacy threats, concerns about maintaining privacy and freedom (which therefore need to be protected from privacy threats), and concerns about things

that may result in privacy threats. The table in Appendix B is the result of grouping these concerns accordingly, and matching them to Solove's Taxonomy of Privacy [threats] [34]. I use the same terminology as [34] in the SIG of Privacy Threats (Appendix C). I illustrate technologies and sources of data that could lead to these privacy threats. I have, however, extended [34]'s Taxonomy of Privacy for information technology as follows:

- Insecurity is a threat that applies throughout the data lifecycle, from the point of collection, to transmission, processing, storage, and data removal. Rather than illustrating it as a subcategory of Data Processing, I illustrate it as a separate type of Privacy Threat.
- I refer to Data Collection, rather than Information Collection, as I consider information to be the interpretation and understanding of the data that is used. Similarly, I refer to Data Processing, rather than Information Processing.
- Indirect data collection occurs in tandem with other data collection. An example of this would be a person in the background of a photograph, or a recording of a discussion in the background, that is captured during the intended recording. I have therefore decomposed Data Collection further, by adding Indirect Data Collection.
- Solove [34] describes Interrogation as a way to seek answers and discusses how inference during interrogation could lead to distortion. Data Mining and Inference of additional information from data are similar in that they also are means of gathering information. They also depend on interpretation of the data that is available and similarly can result in Distortion. However, they are data processes rather than strictly means of gathering data. Also, [15] indicates that inference and derivation are alternate names for the results of data mining, aggregation, or analysis. Since [34] includes

Aggregation as one of the possible threats that results from Data Processing, here, I add another Privacy Threat subgoal for DataProcessing.Derivation. I show Inference as an operationalization contributing positively towards DataProcessing.Derivation, and Data Mining and Data Analysis as alternative operationalizations contributing towards Inference. Since the Data Aggregation operationalization contributes explicitly towards the Privacy Threat DataProcessing.Aggregation, I show Data Aggregation as an implicit contribution towards DataProcessing.Derivation. Similarly, I show Inference contributing implicitly towards the Interrogation and Distortion Privacy Threats.

- The examples [34] provides for Increased Accessibility are of making data that is already public information, more easily available to view and aggregate, thereby facilitating dissemination. However, data persistence may also contribute to having data more "accessible". Therefore, I illustrate DataPersistence as another decomposition type of a Privacy Threat.
- I add Encroachment on Freedoms as an Invasion Privacy Threat, since it does not fit with other types of privacy threats. Larus et al. [51] raises the concern about using machine learning for social engineering. I show Social Engineering as an operationalization of Encroachment on Freedoms. It is related to Intrusion since it influences people's decisions such as political decisions. It is related to Decisional Interference since it influences autonomy in decision-making. Solove [34], however, limits Decisional Interference to private decisions regarding one's body and humanity i.e. one's autonomy. Since it is a type of Invasion, I have included another subcategory, "Encroachment on Freedoms".

The SIG for Privacy Threats in Appendix C shows the privacy threats and the data that is associated with the privacy threats. Note that most of the contributions are shown as S+ since the operationalizations contribute to various degrees towards the privacy threats. Note also, that these privacy threats work ***against*** the privacy softgoal "Avoid Privacy Threat". This means that if any one of the "Privacy Threat" subgoals is satisfiable, "Avoid Privacy Threat" is ***deniable***. Besides privacy concerns, the literature also includes other concerns. Since privacy concerns are balanced with other concerns, I extend the table in Appendix A, to obtain Appendix D. These additional concerns are grouped into business concerns, data subject rights, legal concerns, social concerns (which will not be considered for PrivateMe), and trust.

Several of the concerns listed in Appendix D are shown in the SIGs as operationalizations. This was either determined from the literature, such as, for example, Automated Decision Making which supports decisions that were previously made without such systems [51]. Otherwise, they are known to be operationalizations from my own experience, for example, Globalization which builds Market Share. The (GDPR) Data Subject Rights SIG is based on [8], [22]; it extends on a SIG that I had prepared to illustrate the GDPR, parts of which were shown at CASCON 2018 [52].

Surveillance is not listed as a concern in Appendix D, however, it is mentioned in [15], [16] as an example of how personal data can be collected indirectly, simply by using technology. Since Surveillance is one of the privacy threats included in Solove's Taxonomy of Privacy [34], the different types of surveillance data are included as operationalizations in the SIG for Privacy Threats (Appendix C).

The SIGs for these concerns are included in Appendices: (GDPR) Data Subject Rights, Appendix E and Appendix F, Data Subject Privacy, Appendix G, Business Concerns, Appendix H, Legal Concerns, Appendix I, and Trust, Appendix J.

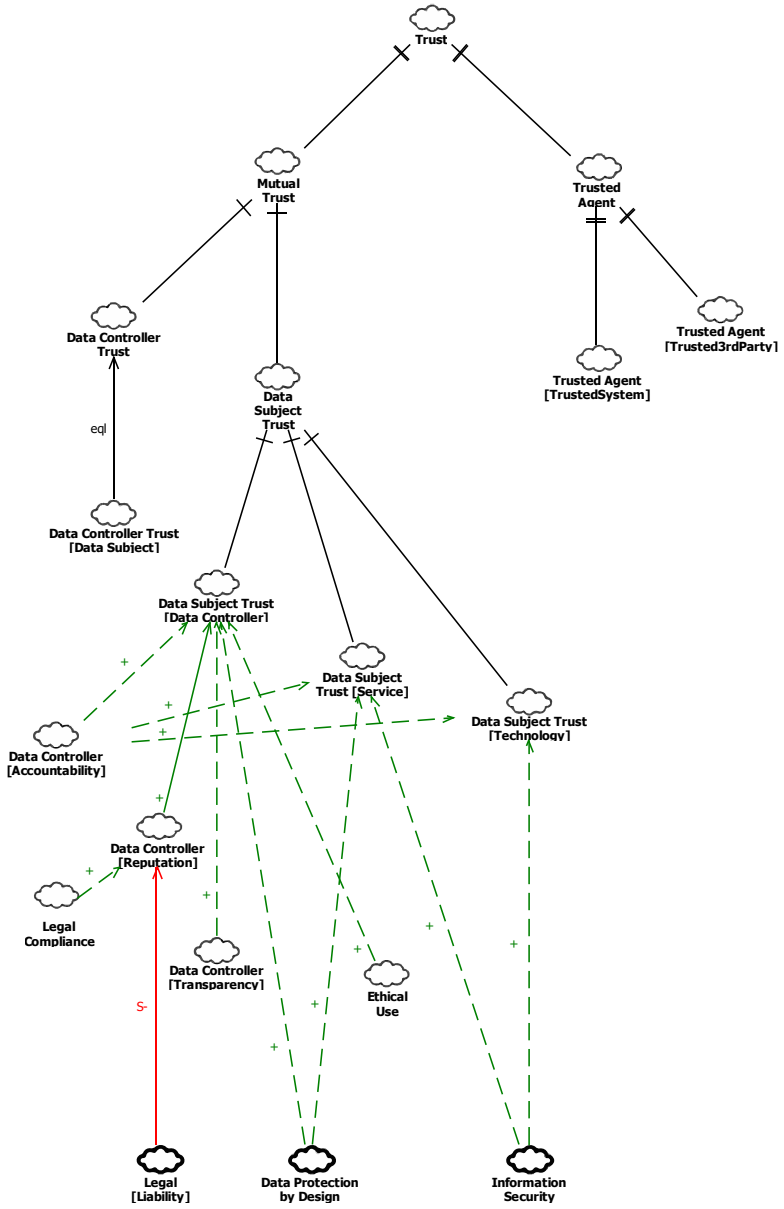


Figure 4: Trust

The SIG for Trust (Figure 4) is included here to illustrate the components of a SIG. Trust can be decomposed into mutual trust (of the data controller and the data subject) *or* (using a) trusted agent [28]. Mutual trust can be further *decomposed* into data controller trust (that is, the data controller's trust of the data subject) *and* data subject trust. This means that both must be *satisficed*

for mutual trust to be *satisficed*. Further decomposition of the data subject trust shows that it includes trust of the data controller, trust of the service, and trust of the technology [19], [28]. The data controller's accountability, reputation, and transparency, as well as ethical use, all contribute positively towards the data subject's trust of the data controller [19], [21]. Since the data controller's transparency, accountability, and ethical use are required by the GDPR [8], they are explicit contributions for compliance with the GDPR, therefore, they provide *correlations* towards the data subject's trust. According to [19], the data controller accountability can *make* or *break* trust. It is, however, one of several contributing softgoals, so, I show it *helps* the data subject trust the data controller, trust the service, and trust the technology. Whereas the data controller's (good) reputation helps the data subject trust the data controller, according to Varadi et al. [21], legal liability is harmful to the data controller reputation. This works against the data subject's trust. Consequently, and as the legal liability can be of differing severities, it is shown with a contribution of *some minus* (S-) towards data controller reputation. Data controller transparency *helps* the data subject trust the data controller. Data Protection by Design is required by GDPR and it *helps* the data subject trust [21]; again, it would be explicit for compliance with the GDPR. So, here I show it has (an *implicit interdependency)* a positive *correlation* towards the Data Subject's trust of the Data Controller and the Service. Similarly, the GDPR requires that the data controller use the data subject's private data ethically (lawfully and by upholding all other GDPR principles [8]), so here, it is shown as (an *implicit interdependency)* a *correlation helping* the data subject's trust in the data controller. Information security is necessary to protect the data controller's information system, also the GDPR holds the data controller accountable for doing so [8]. Information security *helps* the data subject to trust the service and the technology [21], [28]. I show them as *implicit interdependencies* since the primary reason for having information security is to protect the data.

Note that I do not show the Data Protection by Design contributing towards legal compliance in this Trust SIG. This is because Data Protection by Design is only one of several requirements of the GDPR. There are other requirements for legal compliance with the GDPR which do not show in this SIG; showing the interdependency in this diagram would suggest that legal compliance is satisficed with just the Data Protection by Design.

The right half of the Trust SIG shows that as an alternative to mutual trust, the data controller and data subject could use a Trusted Agent. This could be either a trusted system [19], [51] or a trusted third party [28]. As long as one of these softgoals is satisficed, the Trusted Agent is satisficed, and therefore, so is the Trust softgoal.

To ensure that the privacy settings and permissions selected by a data subject, correctly reflect the data subject's privacy goals (RQ1), it is necessary to understand the privacy softgoals of the data subject and to balance them with the benefits of the service offered by the data controller. The privacy concerns gathered from the literature are presented in PrivateMe with the SIG for Privacy Threats. It includes the data that can be used to realize these privacy threats which therefore needs to be protected to avoid the privacy threats. The service provided by the data controller will be of some benefit to the data subject but it will need data to realize the benefit. This can be balanced against the data that needs to be protected to avoid privacy threats. The trustworthiness of the data controller and the service also needs to be considered. Thus, the SIGs in PrivateMe help the data subject to understand what data is necessary to realize a benefit of the data controller's service offering. With this, the data subject can understand how it may impact privacy softgoals. The data subject can consider trusting the data controller and the technology. The data subject can then decide what data needs to be protected and the privacy settings can be selected accordingly.

3.2. Taxonomy

For a data subject to determine whether there are any conflicts between privacy settings and permissions (RQ2), it is necessary for the data subject to know where to find these settings, and to understand what the settings control. Likewise, for a data subject to manage the privacy settings on multiple devices, applications, and web services (RQ3), it is necessary for the data subject to know where to find the settings on devices with different operating systems and to understand how they are similar or how they differ. So, it is helpful to classify what is known about the privacy settings and permissions. The taxonomy in PrivateMe is a classification of the privacy settings and permissions that are available on multiple devices, applications, and websites. In this section, I first explain what a taxonomy is. I briefly discuss the scope of the taxonomy in PrivateMe. I discuss the privacy settings and permissions, as well as the similarities and differences between them, based on how they are accessed and what they control. I conclude this section with a description of the taxonomy that is used for the ontology in Section 3.3.

A taxonomy is a classification system [53], which places real objects into categories or groups, based on some criteria or common features. The criteria for determining how to group the objects is subjective as it depends on what one selects as the defining features of the category or group. This means that there are multiple ways to classify the objects. For example, when classifying two-dimensional shapes, we usually classify them according to the number of edges. In this way, we classify the shapes as triangles if they have three sides, quadrilaterals if they have four sides, and so on. However, we could easily classify the shapes based on colour instead.

The categories or groups of a taxonomy classification system are also known as classes. The objects in a class can be classified into more specific classes, known as subclasses or child classes. The more general class can therefore also be referred to as the superclass or parent class. This

results in a class hierarchy. For example, a Quadrilateral class, the superclass, may have the subclasses Square and Rectangle. Instances or members of these classes, are still quadrilaterals but a square differs from a rectangle in that all four of its sides are the same length.

The classification is abstract since the objects are not actually physically being placed in groups. As an abstraction of the real thing, the taxonomy is a model of the objects within a particular area of interest, which is known as the *domain of discourse*.

I use Protégé [54] to model the taxonomy as well as to build the ontology which is described in more detail in Section 3.3. Whereas the taxonomy is the classification of the privacy settings and permissions, the ontology adds information about the associations between the different classes in the taxonomy. Additionally, the ontology contains the knowledge about how the settings and permissions on each device work together. According to Noy and McGuinness [55], the scope and the anticipated use of the ontology for which the taxonomy is being created, dictates how the domain of discourse is modelled. The purpose for the taxonomy in PrivateMe is to classify the privacy settings and permissions on different devices. Therefore, the domain of discourse is the privacy settings and permissions on multiple devices. The scope includes the privacy settings and permissions, the associations between them, and whatever they control, such as access to the location or to private data.

To create a taxonomy, it is necessary to have an understanding of the similarities and differences between the objects in the domain of discourse. Whereas the similarities are used to define classes, the differences are used to distinguish one class or subclass from another. The privacy settings and permissions include: the privacy settings that are available on a device, the application settings, the application permissions, the website settings, and the website permissions. I describe each more fully in the following paragraphs.

The device privacy settings are part of a larger group of operating system device settings that are used to configure how the device works. They manage things such as "flight mode" and "ring tones". For PrivateMe, however, I limit the scope to the device privacy settings. The device privacy settings control access to private data on the device. They supersede permissions requested by applications. As such, they need to be included in the taxonomy for PrivateMe, so that the ontology can store knowledge about how they manage this control. An example of a device privacy setting (on all devices) is the Location device setting, which controls whether or not any application can access the location of the device.

The application (app) settings control the configuration of an app. Some app settings are available and set from within the app, others are available with the device settings. Examples of app settings are the account setting for an email client, the source of the Contacts list, and the sort order of the Contacts. I include app settings in the PrivateMe taxonomy since this is where the data subject can set up or access an account or profile. When the data subject is signed in to the app, private data such as the data subject's activity history and use of the app may be associated with the account or profile. Browser settings are application settings for the browser application. These settings configure the browser, for example, to automatically sign in for the data subject, to open with a specific web page, or to specify the default search engine. The website (site) settings are similar to app settings but they are applicable to a website.

The application (app) permissions are the permissions that applications display to request access to other apps or to private data on the device, such as the location. The app permissions differ from app settings because they are used to request access to other apps or to data that is external to the app rather than being used to configure the app itself. I include the permissions in the PrivateMe taxonomy because of the request to access private data. Website (site) permissions

are accessed and managed through the browser settings. They do not, however, affect the browser. Site permissions request access to other apps or to data such as the location of the device. As with application permissions, the site permissions control what the site is permitted to access. They are similar to app permissions but they are applicable to a website.

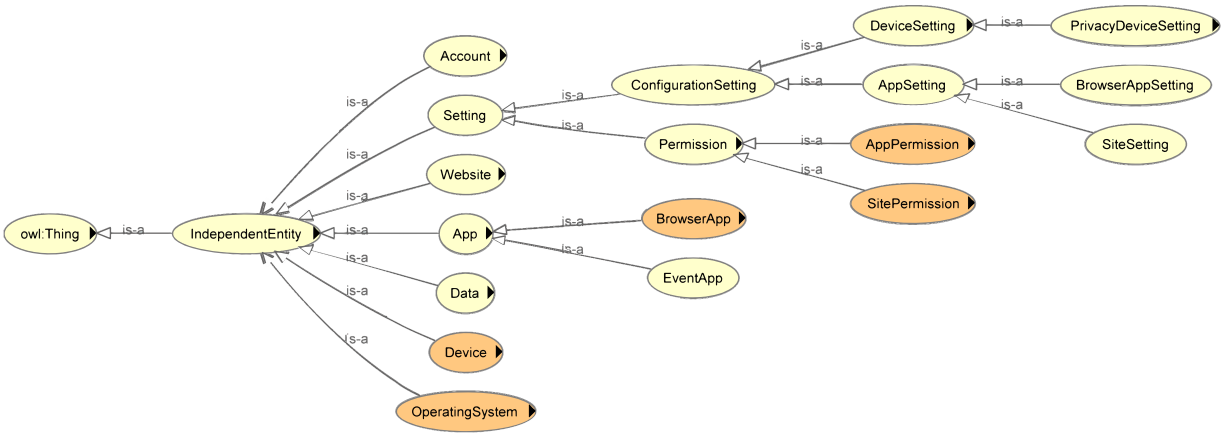


Figure 5: Generic Taxonomy of Privacy Settings and Permissions

The PrivateMe generic taxonomy of privacy settings and permissions is shown in Figure 5. In the taxonomy, the settings and permissions are organized into subclasses of a superclass called Setting, simply because they can all be "set". They are typically accessed through an icon or a menu item named Settings. The device privacy settings, the application settings, the browser settings, the website settings, the application permissions, and the website permissions are the subclasses in the taxonomy. The settings and permissions can be grouped together and classified according to the similarities in what they control. There are settings that control configuration and there are settings that control permissions. The taxonomy therefore includes the Setting subclasses: ConfigurationSetting and Permission.

The ConfigurationSetting class has the subclasses DeviceSetting and AppSetting since these settings configure the device, or are used to configure the app itself. Note that I have mentioned

device privacy settings in previous paragraphs to refer to privacy settings that are on the device. In the taxonomy, this group of settings is called the PrivacyDeviceSetting to follow a naming convention such that the parent class is indicated in the suffix of the class name [55]–[57]. The PrivacyDeviceSetting class is a subclass of DeviceSetting. As previously mentioned, the browser settings are app settings for the browser app. Therefore, the BrowserAppSetting is a subclass of the AppSetting class. The AppSetting class also has the subclass SiteSetting, since SiteSetting is used to configure a website (app) in the same way as an AppSetting is used to configure an app.

Permissions control what the app or the site is permitted to access. The Permission class has the subclasses AppPermission and SitePermission. Site permissions are more similar to the app permissions than to the browser settings, even though they are accessed and set in browser settings. Noy and McGuinness [55] recommend that the classification be based on "intrinsic properties", that is, based on the characteristics of the settings [58]. Therefore, the SitePermission class is classified as a subclass of the Permission class, rather than as a subclass of the BrowserAppSetting, as shown in Figure 5.

As mentioned previously, each operating system manages settings and permissions somewhat differently. The differences become more evident in the details of how the settings and permissions are accessed and how they interact. These details form the knowledge that is captured in the ontology. I continue to add classes and subclasses to the taxonomy discussed here (Figure 5) to build the knowledge for the ontology in Section 3.3.

3.3. Ontology

To ensure that the privacy settings and permissions support the data subject's privacy goals (RQ1), it is necessary to know how the settings and permissions interact. Similarly, to ensure that

there are no conflicts between them (RQ2), it is important to understand the overall result of the settings and permissions. To manage the privacy settings and permissions on multiple devices, for multiple applications and websites (RQ3), it is helpful to understand the similarities as well as the differences between them. The ontology in PrivateMe captures such knowledge about the privacy settings and permissions. Each operating system has a different path to access the settings. The permissions requested by the applications, also differ depending on the operating system. There are, however, some common aspects of the settings and permissions, which I cover in this section. The differences are more evident and are covered with the use cases in Chapter 4. I begin with the reasons for using an ontology and what it can offer, that is not available in a class diagram or in an entity relationship diagram (ERD). I discuss the components of the OWL2 Web Ontology Language [59] (OWL2) that I use to create the PrivateMe ontology. I provide an example of how the ontology can be navigated to check the privacy settings and permissions. I conclude this section with an explanation of how the SIG, the taxonomy, and the ontology in PrivateMe addresses the research questions.

When creating an ontology, objects in the domain of discourse are organized into a taxonomy of classes [55], [57], [60]. The classes are sets of class members [57], [60]. The object properties associate the individuals of one class with individuals of other classes [55], [60], [61]. These features of the ontology are similar to other models such as a UML class diagram [62] or an Entity Relationship Diagram (ERD). However, there are other features of an ontology that distinguish it from either of these types of models and make it more appropriate for sharing knowledge.

OWL2 is used for ontologies which are made available as semantic web documents [59]. The Merriam-Webster dictionary defines “Semantic” as the meaning of language [63]. The Cambridge Dictionary defines “Semantic Web” as storing electronic information on the World Wide Web

such that it is accessible and readable by machine [64]. The semantics add meaning to the structure of the ontology [59]. Therefore, a semantic web document is a document that contains information. The document has a meaningful structure, is machine-readable, and it can be shared via the World Wide Web. Consequently, an ontology creates a common vocabulary to be used in the domain of discourse [55], [59], [60]. OWL2 uses logical statements to describe objects and their associations, within the domain of discourse [55], [59], [60]. Reasoners are computer programs that can be used to check these logical statements (axioms) to ensure that they are consistent, that is, that there are no contradictions between the statements which would result in an empty set [60], [61]. OWL uses open-world reasoning which means that even if something is not stated, it is still possible - unless there are other statements to the contrary [60], [61]. Therefore, reasoners can also infer information about the objects that is not stated explicitly [57], [60]. The ontology makes it possible for the knowledge to be shared and reused, and it makes tacit knowledge explicit [55], [60], [62]. Therefore, the main reasons to use an ontology are 1) to share knowledge via the Web, 2) to create a common vocabulary, 3) to create a document that has a meaningful structure so that it is machine readable, 4) to check for consistency, and 5) to make tacit knowledge explicit. Another reason for using an ontology is to separate domain knowledge from that of business processes [55]. I discuss this more fully below.

Class diagrams and ERDs, are used to design solutions to support functional requirements. In an ERD, an entity is something in the domain of discourse, about which data is retained (for example, in a database). Each entity in an ERD is related to one or more other entities. The relationship in an ERD describes the association between two or more entities, whilst the cardinality indicates the minimum and maximum number of the associated entities. A UML class diagram is used to design the classes that are used in object-oriented application development. The

classes are arranged hierarchically such that each subclass is more specialized than its superclass. Class diagrams include attributes of the class as well as methods which describe the class functionality. The class diagram also associates a class with other classes. These associations have a cardinality to describe the number of associations an instance of the class can have with instances of the associated classes. The class is instantiated as the program is running. According to Hitzler et al. [60], in an ontology, an entity refers not only to the classes, but also the object properties, the data properties, and even statements that form new sets of objects. Thus, an entity in an ontology, has a broader meaning than an entity in an ERD. Also, the classes in an ontology differ from UML classes since functional methods are not part of the ontology. Thus, the domain knowledge in the ontology is separated from the business processes.

In the next few paragraphs, I discuss components of the OWL2 language that I use for the PrivateMe ontology. I briefly discuss the naming convention that I use and the reasons for the exceptions to the naming convention. Then, I discuss the use of primitive and defined classes in the ontology. This is followed by a discussion about axioms, which capture the knowledge of how the settings and permissions interact to manage privacy. In Table 2, I show how the ontology can be navigated. As mentioned in Section 3.2, I use Protégé [54] to create the ontology. I use the HermiT reasoner, and I use OWLViz for the ontology graphs; both are available in Protégé.

I follow a naming convention for the ontology as recommended in [55], [57]. There are, however, some exceptions, especially if it is more indicative of what is normally used. Therefore, the operating systems do not have "OperatingSystem" as a suffix and the iOS operating system begins with an initial lowercase letter. Although [55] recommends against using abbreviations, I abbreviate some names to avoid overly lengthy names once the suffix is added. For example, I use . . . LocPrivDevSetting rather than . . . LocationPrivacyDeviceSetting.

The PrivateMe ontology uses the reasoner to determine whether the settings and permissions allow an App or a Site to access private data. It does so by reasoning about class membership. In an ontology, a subclass only states that it is necessary for a member to be a member of the parent class; this, however, does not mean that a member of the parent class is definitely a member of the subclass [60]–[62]. A class with only necessary conditions is called a *primitive class* [57], [62]. If the class axioms that describe the subclass are enough for an instance, or an *individual*, to definitely be a member of the subclass, then the axioms are necessary and sufficient conditions [61], [62]. As mentioned in Schreiber [62], in OWL, the axioms can define an *equivalent class*. According to Horridge [61], in Protégé the axioms that define the equivalent class can be used to change a primitive class to a *defined class*; the reasoner can then infer additional information about the class. Individuals can belong to more than one class, unless the classes are mutually exclusive, or *disjoint* [57], [60]. The recommendation in [61], however, is to model subclasses with only one superclass, and to allow the reasoner to infer what is inherited from other classes. Therefore, in the PrivateMe ontology, the reasoner infers whether an instance of one class is also a member of the defined class.

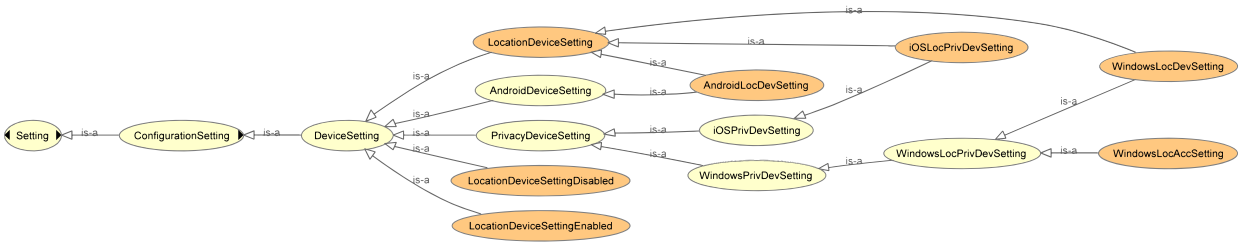


Figure 6: Ontology graph of DeviceSetting class showing settings inferred as LocationDeviceSetting

Figure 6 shows how although the path to access the Location device setting is different for each operating system (AndroidLocDevSetting, iOSLocPrivDevSetting, or WindowsLocDevSetting), each one is also inferred to be a LocationDeviceSetting. The reasoner infers which individuals of these subclasses also belong to the defined classes (indicated by the colour orange) LocationDeviceSettingEnabled and LocationDeviceSettingDisabled.

The operating system names are trademarks of their respective owners. Typically, these would not be concatenated with other words, nor would they be shortened [1], [3], [65]. In the ontology, however, they are class names. According to [55], Protégé allows spaces in the class names, however, it also depends on where else the ontology may be used. Here, since the goal is to make the ontology reusable, I use class names that are a single string. Each class name is used to describe the operating system that manages a particular type of privacy setting. The operating system name is concatenated with other descriptive wording to follow the naming convention mentioned previously, that is, to indicate the parent class.

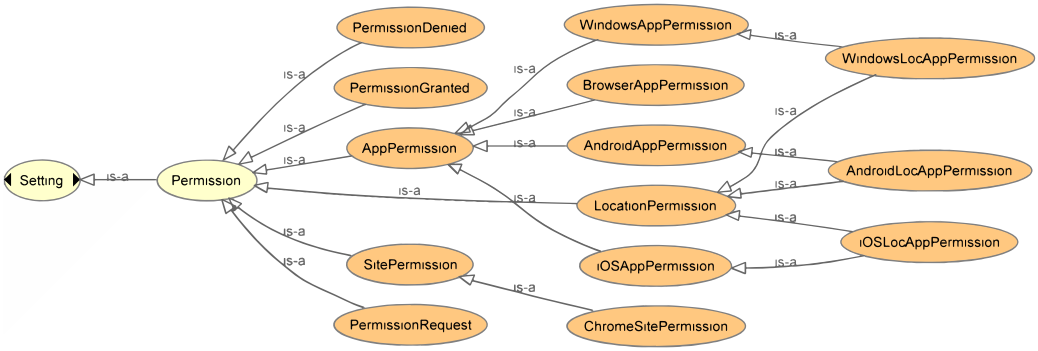


Figure 7: Ontology graph of Permission class showing settings inferred as LocationPermission

Figure 7 shows the AppPermission classes that are inferred to be LocationPermission subclasses. The reasoner infers which individuals of these classes belong to the PermissionDenied, PermissionRequest, or PermissionGranted classes.

McCrae and Unger [66] indicate that an OWL2 axiom for a property is expressed as a triple: a subject, a predicate, and an object. In [56], the class to which the subject belongs is called the *domain*, the predicate is the *property*, and the class to which the object belongs is called the *range*. An *object property* associates individuals of a domain class with the individuals of the range class [55], [57], [60]. A *datatype property* associates individuals of the domain class with data of the specified XML schema data type [55], [57], [60].

An object property in OWL2 may have an *inverse property*. This means that if the domain class D is associated with the range class R, via the property P, then R is associated with D with the specified inverse property P_I [57]. Most of the object properties in the PrivateMe ontology have an inverse so that the property can be navigated via the inverse. An example of this in the PrivateMe ontology is hasApp (which is used with the Device class) and its inverse isAppOnDevice.

Settings have values assigned to them. So, it is possible to use datatype properties to associate the Setting subclasses with the appropriate values. Some of the settings have Boolean values (On/Off). Others, such as the values for the permissions, specify conditions such as "While Using", or "Ask". These values differ depending on the device operating system. In the case of a website, the permissions available depend on the browser that is used. Datatype properties can also be used for the permission values. In that case, the possible values could be listed (*enumerated*) [55]. There are situations, however, where the literature recommends using classes for the values in the range. The domain classes are then associated with the values via object properties, rather than via datatype properties. For example, if the values are to be used as restrictions for other classes, or if individuals in the domain class are distinguished from each other because of their values [55]. Uschold [67] also states that since a datatype property does not have

an inverse, it cannot be used as a subject in a triple, and suggests that in that case, it may be better to use a class for the values. Rector [56], and Horridge [61] show how this is done by using a *value partition* and *covering axiom*. I use a value partition for the PermissionValue class. This is so that the values can be used to classify the applications and sites according to their permissions. The PermissionValue class includes On and Off permissions so that they can be handled in the same manner as the other possible values for permissions.

Table 2 lists the ontology classes, the object (or datatype) properties, and the associated range classes in sequence, to show how the individuals of the classes are associated with each other, as well as what the reasoner infers about the individuals. The ontology can be navigated in this manner to determine whether an app has access to the location data. In the example, I focus on the location of the device as it requires the privacy setting to be enabled, as well as the permission to be granted, before an app or website can access the location. (Statements for equivalent classes have been omitted for brevity but they can be exported from the ontology in Protégé.)

Table 2: Classes and Object Properties for Location Device Setting and App Permissions

	Domain Class	**Object Property** **(Datatype Property)**	**Property Restriction**	**Range Class**
1.1	Device	hasOperatingSystem	some	OperatingSystem
1.2	OperatingSystem	hasDeviceSetting	some	DeviceSetting
1.3	. . . LocDevSetting	enablesAccessToData	exactly 1	LocationData
1.3.1	. . . LocDevSetting (**inferred** subclass of LocationDeviceSetting)	enablesAccessToData	exactly 1	LocationData
1.4	. . . LocDevSetting	isEnabled (DataType Property inherited from DeviceSetting class)	some	xsd:boolean
1.4.1 a **or**	LocationDeviceSettingEnabled (individual setting has **inferred** class membership)			
1.4.1 b	LocationDeviceSettingDisabled (individual setting has **inferred** class membership)			

	Domain Class	**Object Property** **(Datatype Property)**	**Property Restriction**	**Range Class**
2.1	Device	hasApp	some	App
2.2	App	hasAppPermission	only	AppPermission (subclass of Permission)
2.3	. . . LocAppPermission (subclass of AppPermission)	permitsAccessToData (Object Property inherited from Permission class)	exactly1	LocationData (subclass of Data)
2.3.1	LocationPermission (individual/subclass permission has **inferred** membership)	permitsAccessToData	exactly1	LocationData
2.4	. . . AppPermission	hasPermissionValue (Object Property inherited from Permission class)	some	. . . (applicable subset of PermissionValue class)
2.4.1 a **or**	PermissionDenied (individual permission has **inferred** membership)			
2.4.1 b	PermissionGranted			

	Domain Class	**Object Property** **(Datatype Property)**	**Property Restriction**	**Range Class**
or	(individual permission has **inferred** membership)			
2.4.1 c	PermissionRequest (individual permission has **inferred** membership)			
2.5	App or Website	aggregatesData (Datatype Property)	some	xsd:boolean
3.1	AppWithAccessToLocation (individual app has **inferred** membership or DesktopApp is **inferred** subclass)			
3.2	LocationAggregation (. . . PrivacyThreat) (individual App or Website has **inferred** membership)			

If the LocationDeviceSettingEnabled is true and the app has PermissionGranted, then the app has access to the location of the device. Thus, it is possible with the ontology, to determine how the privacy settings affect the data that is captured by a data subject's applications and web services.

Therefore, to ensure that the privacy settings and permissions selected by a data subject, correctly reflect the data subject's privacy goals (RQ1), the SIG captures the privacy softgoals. The taxonomy classifies the device privacy settings and the application permissions that are used to manage the data. The ontology stores the knowledge about how these settings and permissions interact to effectively manage the data. It is necessary to mention, that the PrivacyThreat class includes *potential* privacy threats such as Aggregation. The ontology includes knowledge about data, such as location data, which is also associated with a potential privacy threat. Since the reasoner in the ontology can infer which apps can access this data and therefore possibly pose a privacy threat, the data subject can ensure that the settings and permissions result in the desired privacy softgoals.

It is important for the data subject to determine if there are any conflicts between the settings and permissions (RQ2). This is because any change to one of these settings, may result in the privacy goals of the data subject to no longer be fully met. A conflict occurs when a change to the privacy settings or permissions for one app on a device, results in a change to the net effect of the combined privacy settings and permissions for another app on the same device. The reasoner in the ontology can infer the net result of the settings and permissions for each app. Thus, the data subject can ensure that as the settings and permissions are selected for one app, the net effect of the settings for the other apps continue to result in the desired privacy.

Finally, to manage the privacy settings and permissions on multiple devices, for multiple applications and websites (RQ3), the ontology can determine if there are any conflicts for the same app on multiple devices, or between the app and the website (also on multiple devices). Therefore, RQ3 is similar to RQ2 but applies to multiple devices, apps and websites. Steps 1.3, 1.4, 2.3, and 2.4 in Table 2, show ". . ." in the Domain and Range classes which can be replaced by the applicable prefix "Android", "iOS", or "Windows". As with RQ2, the ontology will show whether there are conflicts between the settings, as well as whether a change to any settings will result in a conflict.

The SIG in PrivateMe shows how the data contributes towards the data subject's privacy softgoals. The taxonomy classifies the device privacy settings, and the permissions for applications and websites. The ontology models how the privacy settings and the permissions interact to manage access to the data subject's private data. By adding to the ontology, knowledge about how the data contributes towards the privacy softgoals, as in the SIG, it is also possible to check that the settings support the privacy softgoals (RQ1). The ontology is also used to determine if there are any conflicts between the settings and permissions (RQ2), and to manage various privacy settings on multiple devices (RQ3).

Chapter 4: Evaluation

I evaluate PrivateMe with one use case per research question. I introduce the use cases by providing a description of a data subject's activities to purchase electronic concert tickets. The data subject's privacy concerns and decisions correspond with the research questions. The PrivateMe technique involves reviewing the SIG to establish the data subject's softgoals and to identify the data associated with those softgoals. By referring to the taxonomy, the data subject is able to find which privacy settings and permissions control the data. The data subject uses the ontology to check how the privacy settings and permissions interact, and to determine whether there are any conflicts between them. This helps the data subject to select the appropriate privacy settings and application permissions to meet the privacy softgoals. By following the PrivateMe technique for the use cases, I show that the PrivateMe SIG, taxonomy, and ontology satisfy the research objectives.

4.1. Use Case Summary

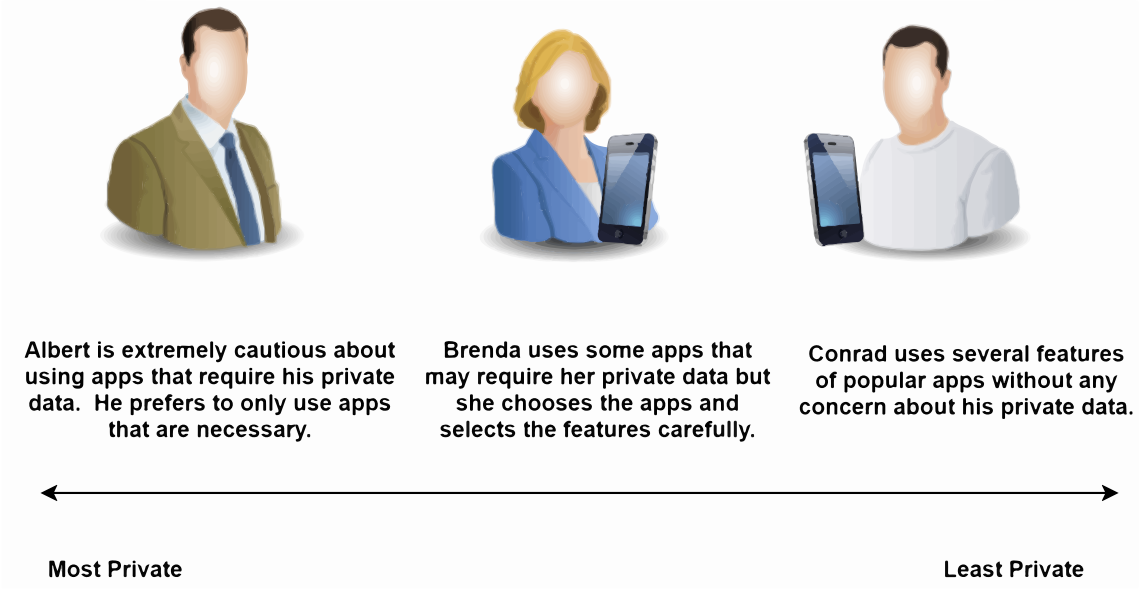


Figure 8: Use Case Data Subject

Brenda (Figure 8) is the data subject in the use case. She is interested in attending an upcoming concert. As a data subject, she makes decisions regarding the privacy settings and permissions for the devices, apps, and sites that she uses to find information about the concert and to purchase tickets. This use case summary describes her actions in sequence and it relates her decisions to the research objectives. The use cases then describe how Brenda uses PrivateMe to assist her in selecting the appropriate settings and permissions to support her privacy softgoals.

Brenda uses her tablet, which is running the Windows 10 Home operating system, and a Chrome™ browser[5] to view the website that is promoting the concert. The site redirects visitors to the Ticketmaster website (site) to view the venues and dates for the concert performances and to purchase tickets. The Ticketmaster site requests access to her location so that it can filter the results of her search accordingly. Alternatively, she may input the actual city or venue where she would like to see the concert. She finds that to purchase tickets, she needs to create a profile. She is not opposed to releasing some information, such as her location, but she prefers to restrict it to people she knows or to companies she trusts; these are her privacy softgoals.

As she browses the Ticketmaster site, Brenda realizes that an app is also available. When she next meets her friends, Brenda could use the app on her mobile phone to tell them about the concert. Then, if she does decide to go to the concert, she can present her electronic tickets at the concert. These are some of the benefits of using the app.

Brenda decides to download the app for her mobile phone which is running the Android™ 6.0.1 operating system[6]. (Hereafter, I refer to the "mobile phone with Android"[7] as such, or simply as *mobile phone*). Once the app is installed, she needs to select the appropriate privacy settings

[5] As requested in [65], this trademark is as in [68]
[6] The trademark is used here as shown in [69].
[7] The term "mobile phone with Android" is used as suggested in [9].

and permissions for her location and her profile. She wants to ensure that the privacy settings and permissions she selects correctly reflect her privacy softgoals (RQ1).

Brenda also wants to make sure that the privacy settings and permissions do not affect the permissions of any other existing apps on her mobile phone. To do this, she must check that there are no conflicts between the settings and permissions that she chooses for the Ticketmaster app and the permissions she has already set for the other apps on her mobile phone (RQ2), such as the Camera app.

The following weekend, when Brenda is out on a boating trip, her mobile phone falls into the lake. Although she manages to retrieve it, she has to replace it. She chooses an iPhone[8] mobile phone[9] which is running the iOS 13.5.1 operating system. She loads the Ticketmaster app on her new iPhone mobile phone. Having the app on her cell phone is convenient but when Brenda decides to purchase the concert tickets, she prefers to do so via the site, simply because her tablet has a larger screen. There is no reason for her to change the trust she has in Ticketmaster (the data controller), so her decision about whether to permit access to her location remains the same. She wants to manage the privacy settings and permissions she chooses on her new iPhone mobile phone such that they match what she had on her mobile phone. She also wants to ensure that she manages the privacy settings and permissions consistently, whether she uses the app on her cell phone or she accesses the site by using the browser on her tablet (RQ3).

The following use cases show how Brenda uses the PrivateMe technique. She uses the SIGs, the taxonomy, and the ontology to select the appropriate privacy settings and permissions to support her privacy softgoals (RQ1). She then uses the taxonomy and the ontology to determine if there are any conflicts between the privacy settings and permissions for the new app and any

[8] iPhone is a trademark of Apple Inc.
[9] The trademark notice above and the generic term “mobile phone” are as requested in [7].

other apps (RQ2), so that she can avoid any such conflicts. Finally, she uses the taxonomy and the ontology to manage the privacy settings and permissions for the app and the site on multiple devices (RQ3).

4.2. Use Case 1: To Ensure that Privacy Settings and Permissions reflect Privacy Softgoals (RQ1)

4.2.1. Use Case 1 Softgoal Interdependency Graphs

Brenda needs to decide whether to permit the app access to her location (data) and whether to create a profile. She refers to the PrivateMe SIGs to help her make these decisions. The SIGs help Brenda identify the private data that is being used by the service and the associated potential privacy threat, which she considers and weighs against the benefits that she receives from the service.

Figure 9 shows the parts of the Business Concerns, Privacy Threats, and Trust SIGs, found in the Appendices, that apply to the use case. It includes operationalizations to represent the Ticketmaster event app and the data that contributes towards the softgoals. It shows the steps that Brenda takes when using the SIGs. These steps are described in the following paragraph and numbered (in parentheses) to correspond with the labels in Figure 9. Although Brenda is navigating the SIGs to select the appropriate privacy settings and permissions for the app, the steps mentioned here also apply to the site.

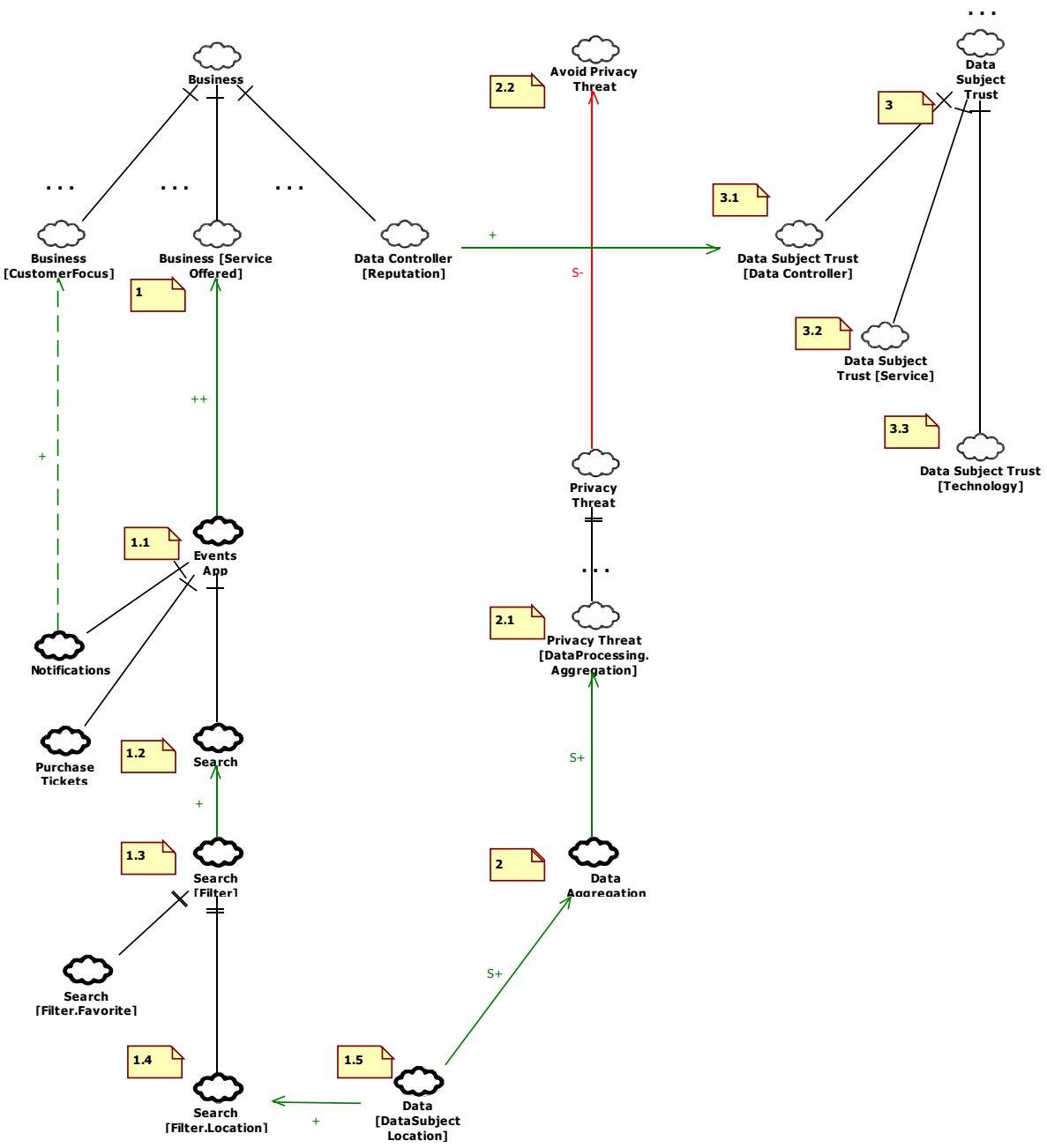


Figure 9: Use Case 1 SIG for Location Permission

Beginning with the Business Concerns SIG (1), the service that is being offered by Ticketmaster is access to view information about events and to purchase tickets for the events (1.1). The app makes it possible to search for events (1.2) and to filter the results of the search

(1.3), for example, by location (1.4). Brenda can either allow access to her location or she can input a location (1.5). If she allows access to her location, it would make it easier for her to find events that are occurring nearby.

From the Privacy Threat SIG, she sees that location data can be aggregated with other data (2) and this could possibly pose a threat to her privacy (2.1). Brenda would like to avoid potential privacy threats (2.2), however, she is not opposed to releasing some information if it is only to people that she knows or companies that she trusts.

Brenda refers to the Trust SIG (3). She must decide whether to trust the data controller, Ticketmaster (3.1), the service (3.2), and the technology (3.3). This refers to Ticketmaster as a company and data controller. The service makes it possible to search for information about the events and to purchase the tickets. The technology supports the service and it includes the app, payment processing, data transmission and storage. Since the data controller's (good) reputation helps trust, it is indicated by the positive contribution towards the data subject's trust in the Trust SIG. Different people base their trust of others upon various factors. For example, it may be based simply upon a general "gut feeling" or a recommendation from a trusted friend. In this case, Brenda may make her decision when she refers to the "About Us" link [70]. She may base her trust on the other companies that are listed as its "Friends & Partners"[70], which she may know and recognize. She may refer to the privacy policy [71] to read about their data security and with whom the data is shared.

As she is satisfied with the information that is provided, Brenda decides that the data controller, the service, and the technology are trustworthy. She also considers how convenient it is when she does not have to input a location, and she weighs this against the potential privacy threat. She decides to grant the app permission to the location data. Thus, Brenda's decision to grant

permission to her location reflects her privacy goal of limiting it to people she knows and companies she trusts (RQ1).

Before Brenda can purchase tickets, she must create a profile. She refers to the PrivateMe SIGs to make sure that her privacy softgoals continue to be supported. The steps she takes are described in the following paragraph. The numbers in the parentheses, correspond with the numbered labels in Figure 10.

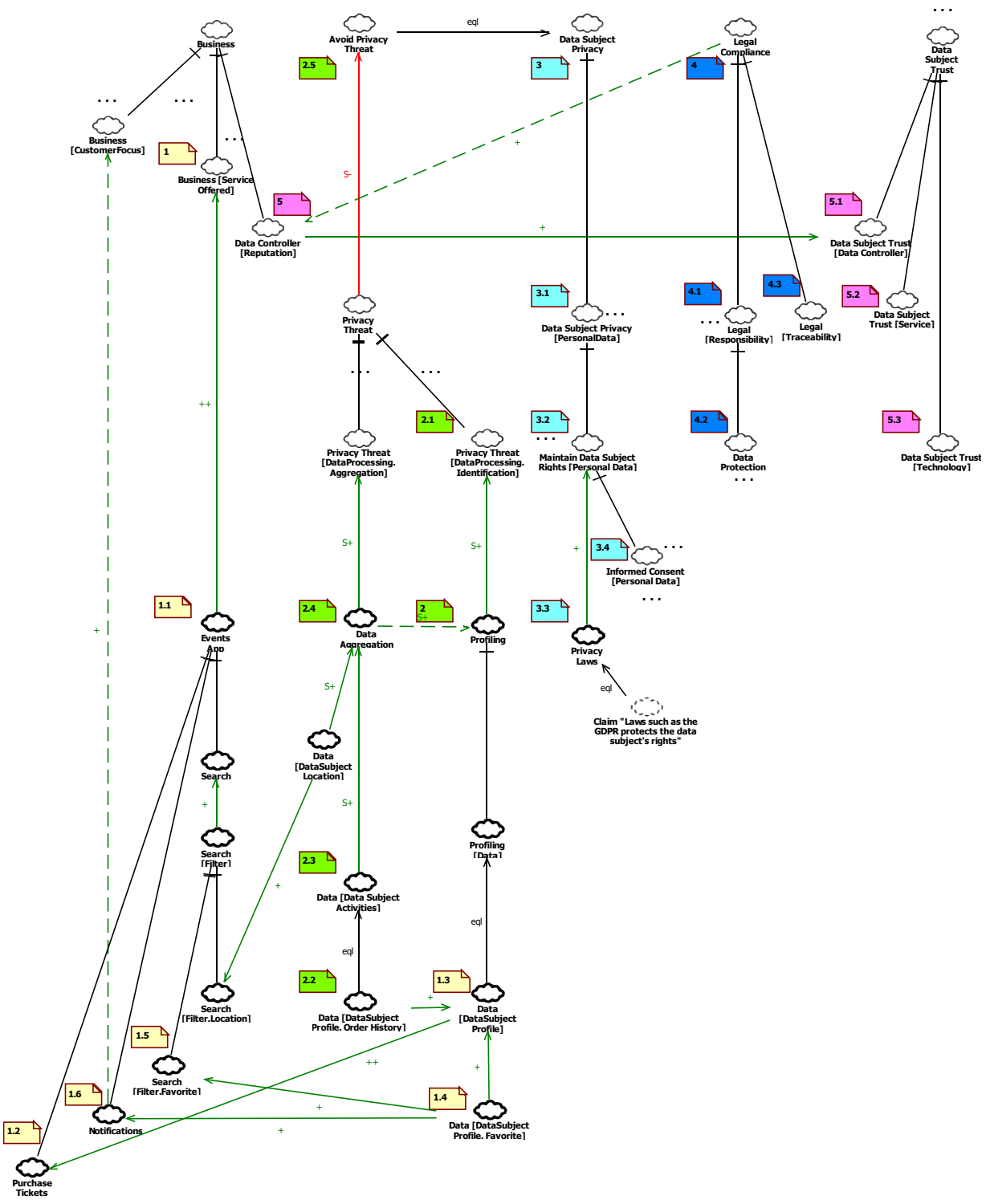


Figure 10: Use Case 1 SIG for Profile

Brenda refers to the Business Concerns SIG. The service that is being offered (1), is realized by the Ticketmaster (event) app (1.1). She can purchase tickets (1.2) but she must create a profile to do so (1.3). Having a profile allows her to store the names of her favourite bands (1.4) so that she can use them as search parameters (1.5). The app can also notify her when upcoming concerts or events feature her favourite bands (1.6).

Brenda's personal data, however, makes it possible for the data controller to profile her (2), which could lead to her being identified (2.1). Similarly, her purchase history (2.2) provides data about her activities (2.3) which can be aggregated with other data. Furthermore, there is a positive correlation between aggregation and profiling (2.4). Brenda would like to avoid privacy threats (2.5) and maintain her privacy.

She refers to the Data Subject Privacy SIG (3) where she sees that to maintain the privacy of her personal data (3.1), she has rights (3.2) which are protected by privacy laws (3.3). One way that these rights are protected is by her informed consent to the use of the data (3.4). As she previously did, before granting permission to her location data, she finds and reads the information that the data controller provides regarding the collection and use of her data.

Brenda sees from the Legal Concerns SIG (4), that the data controller has a responsibility (4.1) to protect the data (4.2). Since her purchase is a financial transaction, however, data for the transaction needs to be retained to maintain traceability (4.3).

From the Trust SIG, Brenda sees that legal compliance implicitly helps the data controller's reputation (5). The data controller's (good) reputation helps her trust the data controller (5.1) but she needs to also trust the service (5.2) and the technology (5.3). Again, as she did previously, she reads the information regarding the services provided. She finds that the Ticketmaster app makes it possible to resell tickets. Brenda decides that although she can trust Ticketmaster, it would be

difficult for her to trust a ticket reseller. As she prefers to limit her information to those she trusts, she makes sure that the tickets she is purchasing are directly from Ticketmaster. She goes ahead and creates a profile, confident that the decision meets her privacy softgoals (RQ1).

By navigating the SIGs in PrivateMe, Brenda associates the service with the data that is being used. She understands the possible privacy threats that can arise from allowing the app to access her location data. She considers her trust in the data controller and the technology, her desire to purchase the tickets, and the convenience of not having to input location each time she uses the site or the app. She balances these against the potential privacy threats to ensure that her privacy goals are maintained. Next, she refers to the taxonomy to find which settings and permissions need to be set to achieve the privacy softgoals. The ontology contains knowledge about the settings and permissions for the app on her mobile phone so that she can set them accordingly.

4.2.2. Use Case 1 Taxonomy

Once Brenda decides to allow access to her location, she needs to select the appropriate settings and permissions to meet her privacy goals (RQ1). She refers to the taxonomy in PrivateMe to determine which settings and permissions she needs to set on her mobile phone, to grant the Ticketmaster app permission to access her location. In this section, I review the settings and permissions that are applicable to Brenda's app and device. I discuss the effect that each one has in enabling access to her location, in Section 4.2.3.

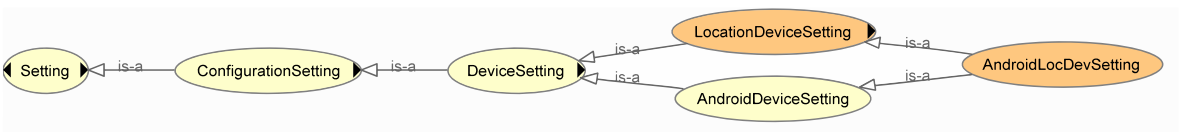


Figure 11: Taxonomy of Use Case 1 App on Android Location Device Setting

As previously discussed, the taxonomy Setting class has two subclasses: ConfigurationSetting and Permission. As shown in Figure 11, the DeviceSetting subclass AndroidDeviceSetting, has a subclass AndroidLocDevSetting. This class is also inferred to be a subclass of LocationDeviceSetting. From this, Brenda ascertains that she needs to use the device setting on her mobile phone to manage access to location.

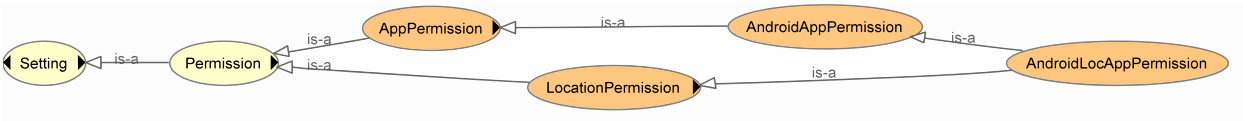


Figure 12: Use Case 1 App on Android Location Permissions

Similarly, as shown in Figure 12, the Permission class has the subclasses AppPermission and AndroidAppPermission. The Ticketmaster app on Brenda's mobile phone requests permission to access the location. This permission corresponds with the AndroidAppPermission, a subclass of AndroidLocAppPermission and also of LocationPermission. This confirms that Brenda needs to set this permission to permit access to her location.

Brenda proceeds to set the location permission in her app. In the following section, I describe the effects of setting individuals of the AndroidLocDevSetting class and the AndroidLocAppPermission class.

4.2.3. Use Case 1 Ontology

By looking at the taxonomy, Brenda can see that there are two classes of settings that affect the access to the location; the AndroidDeviceSetting and the AndroidAppPermission. Within Protégé, classes are instantiated as individuals. The individuals have object properties and datatype properties which relate the individuals to each other. For Use Case 1, the individuals

created in the ontology represent Brenda's mobile phone, the Ticketmaster app, and their settings and permissions. By expressing the properties as axioms relating the individuals to each other, the reasoner in Protege can show the net effect of adjusting the settings on the actual device. Thus, Brenda uses the ontology to ensure that her privacy softgoals are met when she selects the settings to permit the Ticketmaster app on her mobile phone to access her location (RQ1).

In Table 3, I show how the individuals for the Use Case are set for the Ticketmaster app to have the necessary permission to access Brenda's location. The steps are numbered, beginning with the mobile phone with Android operating system and device setting.

Table 3: Ontology of Use Case 1 Individuals

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.1	AndroidPhone (MobilePhoneDevice Device)	APhone	hasOperatingSystem	Android (OperatingSystem)	Android6.0.1
1.2	Android (OperatingSystem)	Android6.0.1	hasDeviceSetting	AndroidLocDevSetting (AndroidDeviceSetting DeviceSetting)	APhoneLocSetting
1.3	AndroidLocDevSetting	APhoneLocSetting	enablesAccessToData	LocationData	APhoneLocData
1.3.1	AndroidLocDevSetting (**inferred** subclass of LocationDeviceSetting)	APhoneLocSetting	enablesAccessToData	LocationData	APhoneLocData
1.4	AndroidLocDevSetting (DeviceSetting)	APhoneLocSetting	isEnabled	xsd:boolean	true
1.4.1	LocationDevice-SettingEnabled	APhoneLocSetting			

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
	(individual setting has **inferred** class membership)				
2.1	AndroidPhone (Device)	APhone	hasApp	EventApp (App)	APhTicketmaster205.0
2.2	EventApp (App)	APhTicketmaster205.0	hasAppPermission	AndroidLocAppPermission (AndroidAppPermission AppPermission Permission)	APhTMLocPermission
2.3	AndroidLocAppPermission	APhTMLocPermission	permitsAccessToData	LocationData (Data)	APhoneLocData
2.3.1	AndroidLocAppPermission (**inferred** subclass of LocationPermission)	APhTMLocPermission	permitsAccessToData	LocationData	APhoneLocData
2.4	AndroidLocAppPermission	APhTMLocPermission	hasPermissionValue	On	on

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
	(Permission)			(PermissionValue)	
2.4.1	PermissionGranted (individual permission has **inferred** class membership)	APhTMLocPermission			
2.5	EventApp (App)	APhTicketmaster205.0	aggregatesData (Datatype Property inherited from App)	xsd:boolean	true
3.1	AppWithAccessToLocation (individual has **inferred** membership)	APhTicketmaster205.0			
3.2	LocationAggregation (Aggregation PotentialPrivacyThreat) (individual App has **inferred** membership)	APhTicketmaster205.0			

In Table 3, the AndroidLocDevSetting class enables access to LocationData. Therefore, it is inferred to be a subclass of the LocationDeviceSetting class. Its class member, APhoneLocSetting, which enables access to the APhoneLocData, is also inferred to be a LocationDeviceSetting. By enabling this setting, (isEnabled is set to true in Step 1.4), it infers that the Location of the device is enabled. This means that by switching the mobile phone Location device setting on, access to the location is enabled.

The app, however, also needs permission to access the location. This is shown in Step 2.3. The APhTMLocPermission permits access to the LocationData APhoneLocData. Again, since it is LocationData, the APhTMLocPermission is inferred to be a LocationPermission. By giving it a PermissionValue of "on", the Permission is inferred to be granted.

The LocationDeviceSetting needs to be enabled and the APhTMLocPermission needs to have PermissionGranted for the APhTicketmaster205.0 app to be able to access the location. Since this is the case, Step 3.1 shows that the APhTicketmaster205.0 app is inferred to be a member of the AppWithAccessToLocation class.

Step 3.2 is added to show that the app aggregates data, which may be a potential privacy threat. Nevertheless, Brenda has already assessed the potential threat and has decided to permit the access to her location. Thus, the privacy settings and permissions that Brenda sets for her mobile phone and for the Ticketmaster app on her phone correctly reflect her privacy softgoals (RQ1).

4.3. Use Case 2: To Determine that there are no Conflicts between Privacy Settings and Permissions (RQ2)

Use Case 2 continues from Use Case 1, in which Brenda enables location access for the Ticketmaster app on her mobile phone with Android. She switches the Location device setting "On" and provides the Ticketmaster app permission to access the location. Although she allows

the Ticketmaster app permission to access her location, she wants to protect her location from being disclosed to other apps. For example, she wants her location privacy to be protected when she takes photos. Therefore, she wants to check how having the Location device setting "On", affects the Camera app on her phone. She refers to the PrivateMe ontology to check if there are any conflicts between the current Location device setting and app permissions, and those which she previously had set for the Camera app (RQ2). Brenda refers to the PrivateMe taxonomy to determine which settings and permissions need to be checked. Then, she refers to the ontology to determine how they affect the Camera app's access to her location. In the next paragraph, I review the Location device setting and the Camera app permissions, as they may have been before Brenda adjusts the settings to enable the Ticketmaster app to access her location. Then, in Section 4.3.2, I list the classes, properties, and inferences for the Camera app on her mobile phone, as modelled in the PrivateMe ontology. I conclude this use case with a brief discussion about the interaction of the Location device setting and the Camera app permissions.

4.3.1. Use Case 2 Taxonomy

Brenda is concerned about protecting her location privacy in her photos. From the PrivateMe taxonomy, she knows that there is an AndroidDeviceSetting, AndroidLocDevSetting, that is also a LocationDeviceSetting and manages access to the device location. There is an AndroidAppPermission, AndroidLocAppPermission, that is also a LocationPermission. As was the case with the Ticketmaster app, she determines that the Camera app also has a location app permission. She determines that she may need to manage the Camera's access to the location via the app permission. For now, I assume that the Location device setting and the Camera app permissions were previously set as shown in the left and the centre screenshots of Figure 13. The

screenshot on the right of Figure 13 is included to show that location tagging in the app is "Off" when the Location device setting is "Off". As mentioned in Use Case 1, however, for the Ticketmaster app to be able to access the location, the Location device setting needs to be enabled and the app permission needs to be granted to the Ticketmaster app.

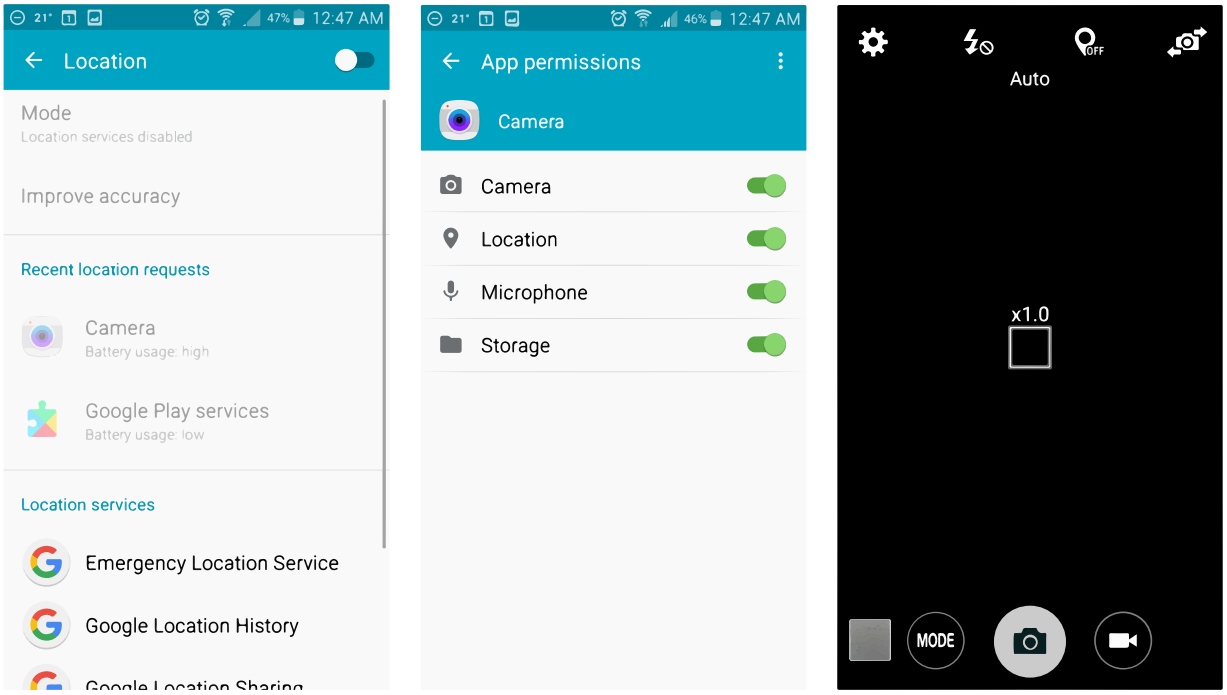


Figure 13: Use Case 2 Location Device Setting and Camera App Permissions on mobile phone with Android

Google and the Google logo are registered trademarks of Google LLC, used with permission [4], [10]–[13].

4.3.2. Use Case 2 Ontology

In Table 4, the APhCameraApp is the individual that represents the Camera app in the ontology; the APhCameraLocPermission is the individual that represents the location permission for the Camera app. All other individuals are the same as those for the Ticketmaster app in Use Case 1. As modelled in the ontology, switching the AndroidLocDevSetting "On" enables the APhCameraApp to access the location, if the APhCameraLocPermission is "On". This is not what

Brenda wants for the Camera app. Therefore, there is a conflict between the settings and permissions for the Ticketmaster app and the Camera app that Brenda needs to resolve. Table 5 shows that, as modelled in the ontology, if the APhCameraLocPermission is "Off" (row 2.4), the reasoner does *not* infer that APhCamera belongs to the AppWithAccessToLocation class (row 3.1). I discuss these interactions between the privacy settings and permissions further in the concluding paragraphs of this section.

Table 4: Ontology of Use Case 2 Individuals

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.1	AndroidPhone (MobilePhoneDevice Device)	APhone	hasOperatingSystem	Android (OperatingSystem)	Android6.0.1
1.2	Android (OperatingSystem)	Android6.0.1	hasDeviceSetting	AndroidLocDevSetting (AndroidDeviceSetting DeviceSetting ConfigurationSetting)	APhoneLocSetting
1.3	AndroidLocDevSetting	APhoneLocSetting	enablesAccessToData	LocationData (Data)	APhoneLocData
1.3.1	AndroidLocDevSetting (**inferred** subclass of LocationDeviceSetting)	APhoneLocSetting	enablesAccessToData	LocationData	APhoneLocData
1.4	AndroidLocDevSetting (DeviceSetting)	APhoneLocSetting	isEnabled	xsd:boolean	true

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.4.1	LocationDevice-SettingEnabled (individual setting has **inferred** class membership)	APhoneLocSetting			
2.1	AndroidPhone (Device)	APhone	hasApp	App	APhCamera
2.2	App	APhCamera	hasAppPermission	AndroidLocAppPermission (AndroidAppPermission AppPermission Permission)	APhCameraLocPermission
2.3	AndroidLocAppPermission	APhCameraLocPermission	permitsAccessToData	LocationData (Data)	APhoneLocData
2.3.1	AndroidLocAppPermission (**inferred** subclass of LocationPermission)	APhCameraLocPermission	permitsAccessToData	LocationData	APhoneLocData

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
2.4	AndroidLocAppPermission (AppPermission)	APhCameraLocPermission	hasPermissionValue	On (PermissionValue)	on
2.4.1	PermissionGranted (individual permission has **inferred** class membership)	APhCameraLocPermission			
2.5	App	APhCamera	aggregatesData (Datatype Property inherited from App)	xsd:boolean	false
3.1	AppWithAccessToLocation (individual/subclass App has **inferred** membership)	APhCamera			
3.2	LocationAggregation (Aggregation PotentialPrivacyThreat) (individual App has **inferred** membership)				

Table 5: Use Case 2 Camera App Permissions

	Domain Class	**Individual**	**Object Property**	**Range Class**	**Individual**
2.4	AndroidLocAppPermission	APhCameraLocPermission	hasPermissionValue	Off	off
2.4.1	PermissionDenied (individual permission has **inferred** class membership)	APhCameraLocPermission			
3.1	AppWithAccessToLocation				

As modelled in the ontology, if the APhLocSetting and the APhCameraAppPermission are both “On”, the Camera app has access to the location. The knowledge captured and modelled in the ontology suggests that to protect the data subject’s location privacy in the Camera app, while the APhLocSetting is “On”, it is necessary to switch the APhCameraAppPermission “Off”. Indeed, this is how the Camera app permissions operate on the iPhone mobile phone and the tablet with Windows devices which are also used in the use cases. The Camera app on the test device is a Samsung SM-A500W, with the Android 6.0.1 operating system and Camera app version 3.0. The App permissions warns that the device will not work as expected, if the Location permission is not granted.

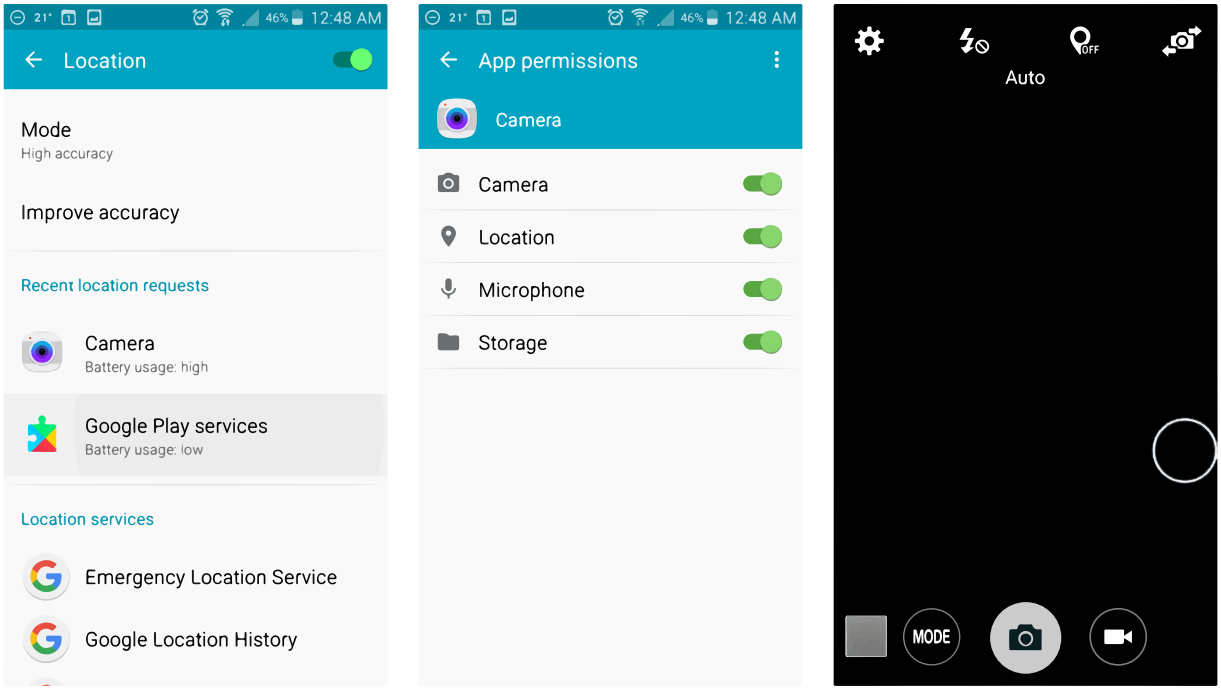


Figure 14: Necessary Location Device Setting, Camera App Permission, and Camera App Setting

Google and the Google logo are registered trademarks of Google LLC, used with permission [4], [10]–[13].

The screenshot on the right of Figure 14 shows the Camera app setting for tagging photos with the location, switched “Off”. Since this is managed from within the app, it is classified in the

PrivateMe taxonomy as an AppSetting, another subclass of the ConfigurationSetting class, along with the DeviceSetting class. So, in the use case, for Brenda to protect her location, she needs to set the Location device setting and the Camera app permissions as shown in the screen shots on the left and in the centre of Figure 14. She also needs to ensure that the app setting for tagging photos with the location is "Off".

Figure 15 shows how the Location device setting, the Camera app permission, and the Camera app setting would appear if the location is to be tagged in the photos. It is worth noting that the Camera app setting on the test device is automatically reset to "Off" when the Location device setting is switched "Off". If the Location device setting is subsequently switched "On" again, however, the app setting for tagging photos with the location needs to be switched "On" manually.

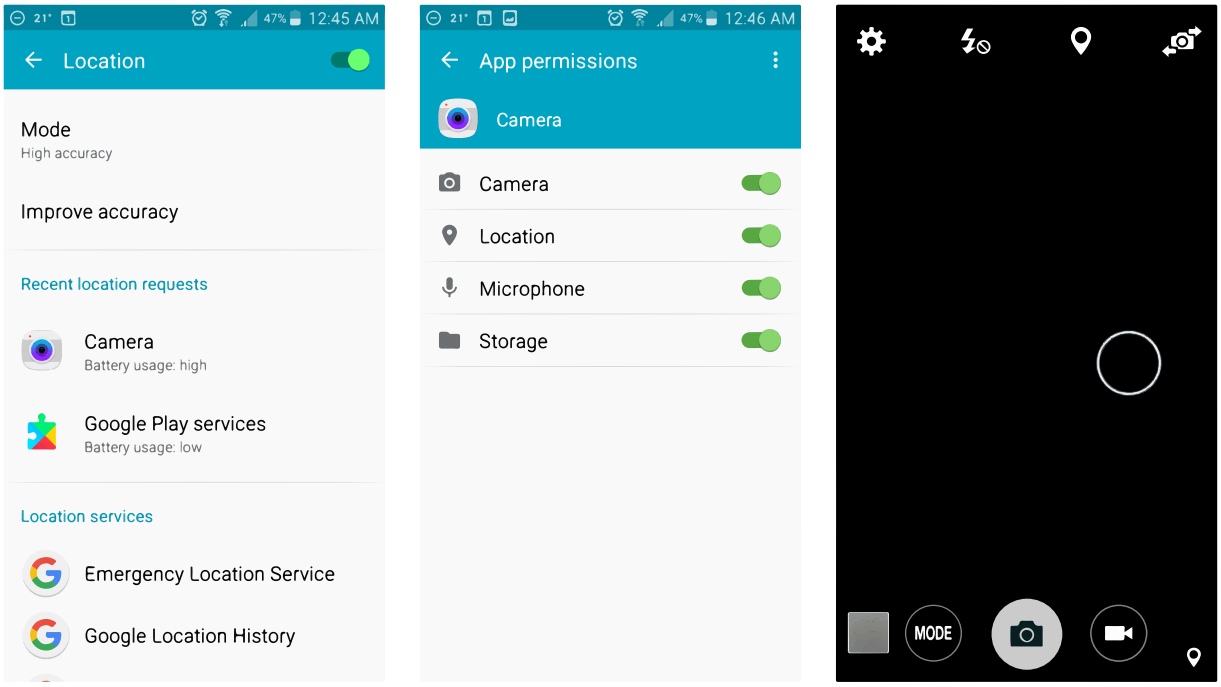


Figure 15: Location Device Setting and Camera App Permissions as Expected from the Ontology

Google and the Google logo are registered trademarks of Google LLC, used with permission [4], [10]–[13].

In this use case, therefore, Brenda can confirm that her location privacy is protected in the Camera app, if she also checks that app setting for tagging photos with the location is "Off". In doing this, she ensures that there are no conflicts between the privacy settings and permissions she has selected for the Ticketmaster app and the Camera app (RQ2).

4.4. Use Case 3: To Manage Privacy Settings and Permissions on Multiple Devices, Applications, and Webservices (RQ3)

In Use Case 3, Brenda wants to ensure that she manages her privacy settings and permissions for the Ticketmaster app and site consistently, whichever device she uses (RQ3). Recall that in Use Case 1, Brenda enables the Location device setting on her mobile phone with Android and she allows the Ticketmaster app permission to access her location. When she installs the app on her new iPhone mobile phone, she wants it to have similar privacy settings and permissions. She also wants to configure her tablet with Windows 10 in a similar manner for the Ticketmaster site. Brenda refers to the PrivateMe taxonomy to find the settings and permissions that she needs to use on each of these devices. Although there are some similarities in how the devices manage the privacy settings and permissions, there are also some differences. The knowledge in the PrivateMe ontology helps Brenda to manage these settings and permissions consistently across devices.

In Section 4.4.1, I review the privacy settings and permissions that are available on each of her devices. I briefly discuss the settings and permissions which Brenda needs to set. The interactions between these settings and permissions are discussed more fully in the ontology which is presented in Section 4.4.2.

4.4.1. Use Case 3 Taxonomy

Brenda refers to the PrivateMe taxonomy to determine which privacy settings and permissions manage access to her device location data on each of her devices. She has Ticketmaster version 205.0 on her iPhone 8, which is running iOS 13.5.1. Brenda accesses the Ticketmaster site on her tablet with Windows via a Chrome browser loaded as a desktop app. (Hereafter, I refer to the tablet with Windows, simply as the *tablet*). She has Chrome browser version 83.0.4103.97 and Windows 10 Home Version 1511, OS build 10586.1176.

In this section, I first review the privacy settings and permissions for the Ticketmaster app on Brenda's iPhone mobile phone. I discuss the similarities to the privacy settings and permissions on her mobile phone with Android (hereafter, *mobile phone*). I briefly describe the Location settings and permissions for an app on Windows 10 Home, since these are also similar in some ways, to the mobile phone and the iPhone mobile phone. Then, I review the Ticketmaster site settings as they are managed via the Chrome browser desktop app on Brenda's tablet.

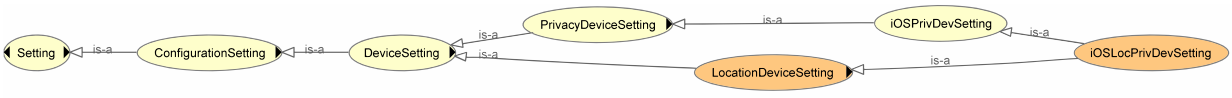


Figure 16: iOS Location Privacy Device Setting

Brenda sees that the PrivateMe taxonomy has a PrivacyDeviceSetting subclass, iOSLocPrivDevSetting, which is inferred to also be a subclass of the LocationDeviceSetting class as shown in Figure 16. Thus, Brenda knows that access to her device location is managed by a Privacy device setting. The iOSLocPrivDevSetting is similar to the AndroidLocDevSetting, in Table 3 and Table 4, however, it is accessed via an additional step; Brenda needs to navigate from

the Settings on her iPhone mobile phone, to the Privacy device settings to find the Location privacy device setting.

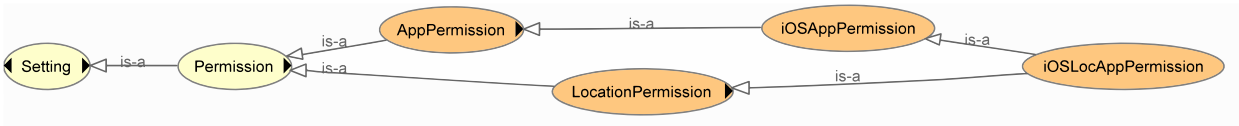


Figure 17: iOS Location App Permission

She also sees that there is a subclass of AppPermission, iOSLocAppPermission, which is inferred to be a subclass of the LocationPermission class, as shown in Figure 17. As she did with her mobile phone, she also needs to grant the Ticketmaster app on her iPhone mobile phone, permission for it to access and use the location.

Figure 18: Windows Location Device Setting

Similarly, the taxonomy shows that there is a PrivacyDeviceSetting subclass, WindowsLocDevSetting, that is inferred to also be a LocationDeviceSetting subclass, as shown in Figure 18. This setting, however, is one of two subclasses of the WindowsLocPrivDevSetting class shown in Figure 19. The additional setting, WindowsLocAccSetting, makes it possible for the primary Location device setting to be managed by an administrator; this secondary Location device setting can then be managed by the individual who is logged in on the device [72].

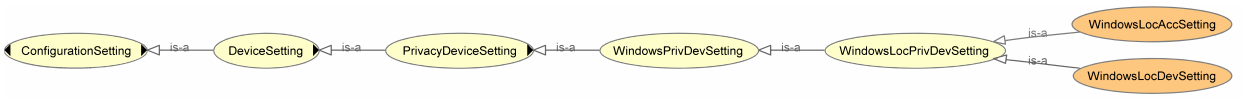


Figure 19: Windows Location Privacy Device Settings

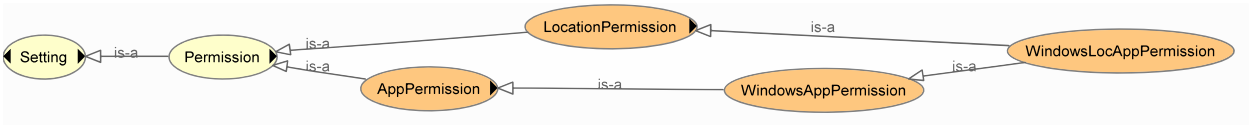


Figure 20: Windows Location App Permission

As shown in Figure 20, there is also a permission class, WindowsLocAppPermission, which is inferred to be a LocationPermission subclass. Again, this is similar to the iPhone and Android app permissions. So, other than the additional setting for the person logged in to the device, the Windows Location device settings and permissions are similar to those of the iPhone mobile phone. The app permissions are also similar to those of the mobile phone and the iPhone mobile phone.

Figure 21: BrowserAppPermission

Since Brenda accesses the site via a browser, she also needs to consider the permissions of the browser. The taxonomy shows the BrowserAppPermission class as a subclass of the AppPermission class as shown in Figure 21.

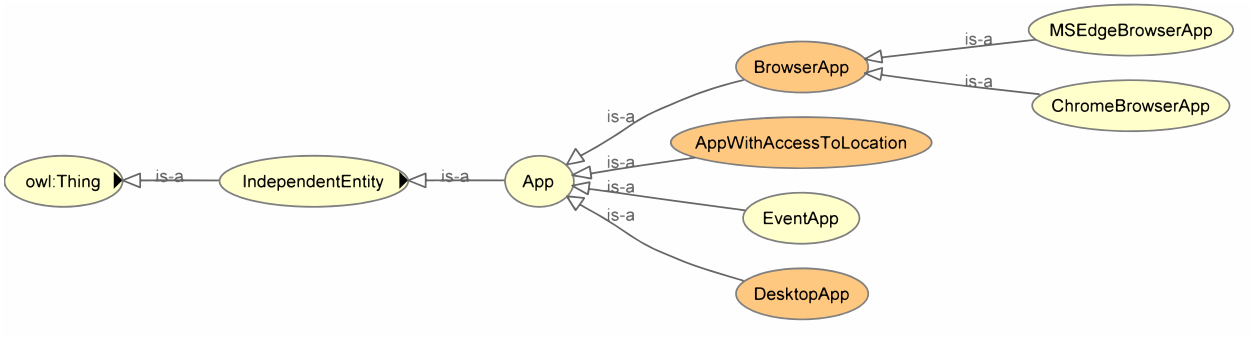


Figure 22: App class showing the BrowserApp subclass

Desktop apps, however, may have access to the device location regardless of these settings [72], [73]. As shown in Figure 22, the ChromeBrowserApp class is a subclass of the BrowserApp class. It is also a desktop app, though, and as shown in Figure 23, a DesktopApp is inferred to be a subclass of the AppWithAccessToLocation class.

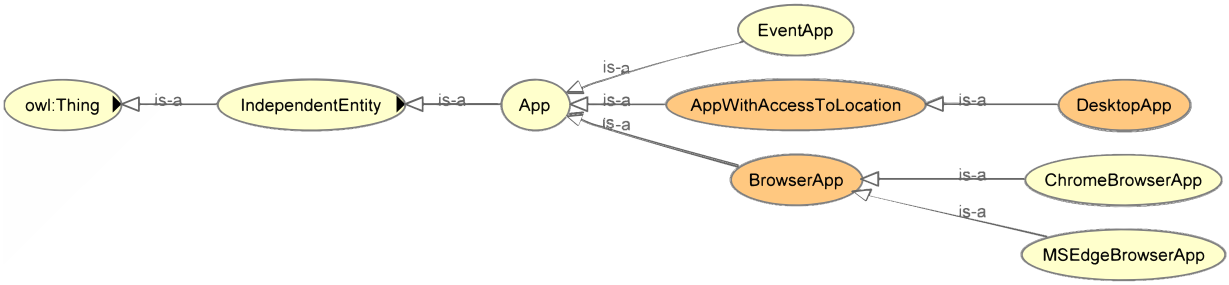


Figure 23: App class showing the DesktopApp as an inferred subclass of AppWithAccessToLocation

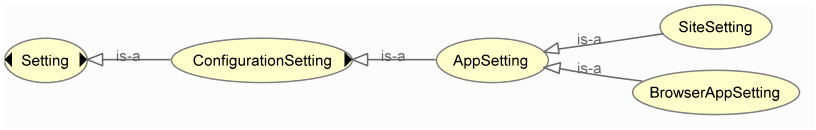


Figure 24: Site Setting and BrowserAppSetting

As shown in Figure 24, the taxonomy distinguishes between SiteSetting and BrowserAppSetting. In this use case, the SiteSetting class would include Brenda's profile and her Favorites. The BrowserAppSetting class manages how the browser operates.

4.4.2. Use Case 3 Ontology

Brenda wants to ensure that she is consistent in how she manages access to her location, whether she is using the Ticketmaster app on her iPhone mobile phone or she is accessing the site on her tablet (RQ3). The knowledge of how the settings and permissions interact is captured in the PrivateMe ontology. So, Brenda refers to the PrivateMe ontology to understand how the privacy settings and permissions work together to manage access to her device location. Table 6 shows the privacy settings and permissions, their object and datatype properties, and the resulting inferences for the individuals that represent the Ticketmaster app on Brenda's iPhone mobile phone. (Here, due to trademark guidelines, I refer to the individual representing the iPhone mobile phone as bPhone, however, this individual is named differently in the ontology.) Table 7 shows the same for the individuals that represent the Ticketmaster site and the Chrome browser on her tablet with Windows.

In Table 6, row3.1, we see that the reasoner in the ontology, infers that iPhTicketmaster205.0, the individual representing the app on Brenda's iPhone mobile phone, does have access to her device location. The TicketmasterSite individual in Table 7, represents the Ticketmaster site as accessed via the Chrome browser (desktop app) on Brenda's tablet with Windows. Table 7, row 5.1, shows that the reasoner in the ontology, infers that TicketmasterSite has access to the location. Thus, the knowledge in the PrivateMe ontology helps Brenda to select the applicable privacy settings and permissions, such that the Ticketmaster app on her iPhone mobile phone and the Ticketmaster site on her tablet, has access to her location.

Table 6: Ontology of Use Case 3 Individuals for iPhone mobile phone

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.1	IPhone (MobilePhoneDevice Device)	bPhone	hasOperatingSystem	iOS (OperatingSystem)	iOS13.5.1
1.2	iOS (OperatingSystem)	iOS13.5.1	hasDeviceSetting	iOSLocPrivDevSetting (iOSPrivDevSetting DeviceSetting ConfigurationSetting)	iPhoneLocSetting
1.3	iOSLocPrivDevSetting	iPhoneLocSetting	enablesAccessToData	LocationData (Data)	iPhoneLocData
1.3.1	iOSLocPrivDevSetting (**inferred** subclass of LocationDeviceSetting)	iPhoneLocSetting	enablesAccessToData	LocationData	iPhoneLocData
1.4	iOSLocPrivDevSetting (DeviceSetting)	iPhoneLocSetting	isEnabled	xsd:boolean	true

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.4.1	LocationDevice-SettingEnabled (individual setting has **inferred** class membership)	iPhoneLocSetting			
2.1	IPhone (Device)	bPhone	hasApp	App (EventApp App)	iPhTicketmaster205.0
2.2	EventApp (App)	iPhTicketmaster205.0	hasAppPermission	iOSLocAppPermission (iOSAppPermission AppPermission Permission)	iPhTMLocPermission
2.3	iOSLocAppPermission	iPhTMLocPermission	permitsAccessToData	LocationData (Data)	iPhoneLocData
2.3.1	iOSLocAppPermission (**inferred** subclass of LocationPermission)	iPhTMLocPermission	permitsAccessToData	LocationData	iPhoneLocData

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
2.4	iOSLocAppPermission (AppPermission)	iPhTMLocPermission	hasPermissionValue	WhileUsing (PermissionValue)	whileUsing
2.4.1	PermissionGranted (individual permission has **inferred** class membership)	iPhTMLocPermission			
2.5	EventApp (App)	iPhTicketmaster205.0	aggregatesData (Datatype Property inherited from App)	xsd:boolean	true
3.1	AppWithAccessToLocation (individual has **inferred** membership)	iPhTicketmaster205.0			
3.2	LocationAggregation (Aggregation PotentialPrivacyThreat) (individual App has **inferred** membership)	iPhTicketmaster205.0			

Table 7: Ontology of Use Case 3 Individuals for Tablet with Windows

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
1.1	WindowsTablet (TabletDevice Device)	WinTablet	hasOperatingSystem	Windows (OperatingSystem)	Windows10Home
2.1	WindowsDevice	WinTablet	hasApp	ChromeBrowserApp (BrowserApp App)	Chrome
2.2	ChromeBrowserApp (BrowserApp App)	Chrome	isDesktopApp (Datatype Property)	xsd:boolean	true
2.3	DesktopApp (individual has **inferred** subclass membership)	Chrome	isDesktopApp (Datatype Property)	xsd:boolean	true
3.1	DesktopApp	Chrome			

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
	(**inferred** subclass of AppWithAccessToLocation)				
4.1	BrowserApp (App)	Chrome	appliesSitePermission	ChromeSitePermission (. . . , Permission)	WinChromeTM-SitePermission
4.2	ChromeSitePermission	WinChromeTM-SitePermission	isSitePermissionFor	Website	TicketmasterSite
4.3	ChromeSitePermission	WinChromeTM-SitePermission	permitsAccessToData	LocationData (Data)	WinTabLocData
4.3.1	LocationPermission (individual has **inferred** class membership)	WinChromeTM-SitePermission	permitsAccessToData	LocationData (Data)	WinTabLocData
4.4	ChromeSitePermission	WinChromeTM-SitePermission	hasPermissionValue	Allow (PermissionValue)	allow
4.4.1	PermissionGranted (individual has **inferred** class membership)	WinChromeTM-SitePermission	hasPermissionValue	Allow (PermissionValue)	allow

	Domain Class (Superclass/es)	**Individual**	**Object Property (Datatype Property)**	**Range Class (Superclass/es)**	**Individual**
5.1	SiteWithAccessToLocation (individual has **inferred** class membership)	TicketmasterSite			
6.1	Website	TicketmasterSite	aggregatesData	xsd:Boolean	true
6.2	LocationAggregation (Aggregation PotentialPrivacyThreat) (individual App has **inferred** membership)	TicketmasterSite			

There are some similarities as well as some differences between the ways the settings and permissions are managed on the devices for Use Case 3. This can be seen from the classes and subclasses in Section 4.4.1, in Table 6, and in Table 7. I summarize these similarities and differences in the paragraphs that follow.

The operating systems on mobile devices (mobile phones and tablets) manage access to the location with a Location device setting. The access is then managed per app by the Location app permission which grants or denies permission to the app. In the following paragraphs, I briefly discuss the similarities and differences between the ways the operating systems that are used in Use Case 3, manage this access.

The Location device setting supersedes the Location app permission. The Location device setting is typically found under Settings. Whereas it can be found directly under Settings on devices with Android, the iOS Location device setting and the Windows Location device setting is found within Settings, under Privacy device settings.

App permissions on Android are binary, that is, either "On" or "Off". Although not discussed for Use Case 3, Windows app permissions are also binary. iOS app permissions are more granular, allowing a choice of "Never", "Ask", "While Using", and "Always". On Windows, if a browser is loaded as an app, it also needs permission to access the device location; when a browser is loaded as a desktop app, however, it may already have access to the device location [72], [73].

SitePermission is similar to AppPermission and is used in this thesis to refer to permissions that allow the site access to data that is on the device (including the device location data). The site permissions, are managed via the browser. The Chrome browser offers the site permission options: "Block", "Ask", or "Allow", they are therefore more granular than the binary options "On" or "Off". In this regard, they are similar to the app permissions offered by iOS. Within the Chrome

browser, however, such SitePermissions are called “Site Settings”. AppSetting and SiteSetting, as defined in this thesis, are similar in that they manage app functionality and site functionality respectively. For the Ticketmaster site, this includes the data subject profile, purchase history, and Favorites. Furthermore, browsers are themselves, also apps, they therefore can have their own set of app permissions and app settings.

Use Case 3 focuses on the Location device setting and app or site permissions. The ontology has been developed to have individuals that represent the use case but as such, it is limited to the Location device settings and permissions for the devices specified here. The Microsoft Edge[10] browser can be loaded either as an app, or as a desktop app [72]. Future work may therefore include developing the ontology to include the Microsoft Edge browser. As mentioned previously, future work may also include developing the AppSetting class. In that case, the SiteSetting class could be developed further as part of that scope of work, since as defined here, the SiteSetting class corresponds with the AppSetting class. In the Ticketmaster app that is used here, this includes data subject profile, purchase history, and Favorites. The caveat is that the app settings are specific to each app and therefore can only be used as an example. Even so, it would help to highlight the differences between the SitePermission and SiteSetting as used for a website. It would also help to distinguish these classes from the AppPermission and AppSetting classes as used for a browser app.

[10] The trademark is used here according to [2]

Chapter 5: Conclusion

Privacy is subjective and complex, and it changes depending on context. A data subject benefits from services offered by a data controller but these services often require the use of private data. This raises privacy concerns about how much private data is being collected, how it is being used, and whether it is being aggregated with other data or used to infer additional personal information. Privacy settings on devices and permissions for apps and sites are provided so that a data subject can manage the disclosure of such data. The settings and permissions, however, interact with each other. This may make it difficult for a data subject to know what to select to effectively control private data. Considering such concerns led to this research. A data subject may want to ensure that the selected privacy settings and permissions meet his or her privacy goals (RQ1). Also, since a data subject may have several apps on a device, he or she may want to ensure that there are no conflicts between the selected settings and permissions (RQ2). Similarly, since a data subject may have more than one device, he or she may want to ensure that there are no conflicts between the settings and permission selected on the devices for the apps and websites (RQ3).

This research contributes PrivateMe, a technique to manage privacy settings on different devices, for multiple applications and web services. PrivateMe includes a softgoal interdependency graph (SIG) which captures the data subject's privacy goals. The privacy threats are also captured in a SIG. The SIG for privacy threats, includes the data operationalizations, to illustrate the type of data that contributes to the possible threats. These SIGs can be used by the data subject to assess the possible threat and to balance it against disclosing the data to obtain the services and functionality offered by the data controller. PrivateMe also includes a generic taxonomy; it classifies the device privacy settings and the permissions that are requested by the

applications and web services. The taxonomy helps the data subject find the settings and permissions that can be set. The last component of PrivateMe is an ontology which stores machine readable and reusable knowledge of how the settings and permissions interact to manage the private data. This research aids the data subject to select privacy settings and permissions that will support the data subject's intentions. As the three use cases show, PrivateMe can be used for multiple devices and for multiple applications and sites.

In this chapter, I discuss the results of this research as well as any additional findings. I also discuss any complications that arose while conducting the research, threats to internal and external validity, and how these were mitigated. I conclude this chapter with a discussion about future work.

5.1. Summary of the PrivateMe Use Case Results

PrivateMe is evaluated with three use cases. The results of the use cases are summarized in the following paragraphs. Additional results are also discussed here as they may lead to opportunities to develop PrivateMe further in the future work.

Use Case 1 shows how PrivateMe can be used to ensure that the settings and permissions support the data subject's privacy goals (RQ1). By navigating the SIGs as shown in Use Case 1, the data subject can relate the data that is being requested (such as Location), to the service that is realized by the app. This will benefit the data subject for example, by filtering the results accordingly. The SIG shows how such data contributes towards a possible privacy threat. This helps the data subject to identify what the possible threat may be. The Data Subject Privacy SIG (Appendix F) shows that the data subject has the right to be informed and to consent to the data collection. This prompts the data subject to find information about why the data is collected, for

example, by referring to the privacy policy provided by the data controller. The decision whether to permit access to the data is also based on trust. Legal compliance makes a positive contribution towards the data controller's reputation (Appendix J); the reputation affects the data subject's trust. Therefore, Use Case 1 shows how the data subject uses the SIGs to ensure that the privacy goals are upheld when deciding whether to permit access to the data (RQ1). The taxonomy helps the data subject to understand where to find the appropriate settings and permissions. The ontology shows how the selected settings and permissions interact to provide the desired level of privacy (RQ1).

Use Case 2 shows how PrivateMe can be used to determine whether there are any conflicts between the privacy settings and permissions for multiple apps on the same device (RQ2). Use Case 2 shows that the Location device setting takes precedence over the app permissions. Thus, to grant the Ticketmaster app permission to the location of the device, the Location device setting must first be switched "On". This change to the Location device setting, however, may affect other applications that may also have previously had the permission. The opposite is also true, that is, if the data subject wanted to remove the Ticketmaster access to the device location, and did so by switching the Location device setting "Off", this change will also affect the other apps that have the permission. Thus, the PrivateMe taxonomy and ontology help the data subject to identify when there are conflicts between the privacy settings and permissions (RQ2).

Use Case 2 provided an opportunity to show the difference between an app setting and an app permission, even though this was not planned. At the beginning of the use case, all three devices had the Camera app with permission to access the location. The Location device setting on all three devices was "Off", therefore none of the Cameras were tagging pictures with the location data. However, upon switching the Location device setting "On" to allow the Ticketmaster app

permission to access location, this enabled the Cameras on the iPhone mobile phone and on the tablet with Windows (*tablet*) to start tagging the pictures. The ontology reasons that the Camera will behave this way. On the mobile phone with Android (*mobile phone*), however, the Camera app setting for tagging the pictures is automatically switched "Off" when the Location device setting is switched "Off". This setting was not automatically reset when the Location device setting was switched "On" for the use case. Thus, Use Case 2 also shows a conflict on the mobile phone, between the Location device setting, the Camera app permission, and the Camera app setting on the mobile phone. The data subject also needs to resolve such a conflict if at a later date, she wants to start tagging pictures with her location.

Use Case 3 shows how PrivateMe can be used to help the data subject manage privacy settings and permissions for the same app or for the site, on multiple devices (RQ3). The taxonomy helps the data subject to find the Location device settings. This is slightly different for each device. There are also some differences between the granularity of the app permissions on the different devices. On the tablet with Windows (*tablet*), instead of using an app, the site is accessed via a browser. Consequently, the site permissions are managed via the browser. The browser may have access to the location or it may also need permission to access the location; this depends on whether the browser is loaded as a desktop app or an app [72], [73]. Use Case 3 (Section 4.4.2) includes a detailed discussion about these differences. Use Case 3 shows how the ontology stores the knowledge about these differences and thus, helps the data subject to manage the settings and permissions for the app on multiple devices, as well as for the site.

There are also some additional findings that result from conducting the use cases. For example, whereas Ticketmaster on Android requests permission to access the Calendar, Camera, Contacts, Location, Phone, and Storage, the corresponding app for the iPhone mobile phone only

asks to access Location, Siri & Search, Background App Refresh, and Cellular Data. These differences may contribute towards the data subject finding it difficult to reconcile permissions to the data that is being collected or the use of that data to the functionality of the app. This may be a topic that researchers may wish to study further in the future.

5.2. Challenges

The operating systems of the devices that were used to evaluate PrivateMe, were set to automatically update, as is typically recommended for security. For example, iOS 13.3.1 was updated during the project to iOS 13.5.1. These updates, however, did not result in a discernable change, neither to the access path for the device privacy settings nor to the way they manage the privacy. Therefore, it was not necessary to update the taxonomy or the ontology knowledge of the device privacy settings.

Similarly, the apps loaded on the two mobile phones were previously also set to automatically update. The updates, however, did not include any changes to the permissions requested by the apps. Nevertheless, the main concern while conducting this research, was to ensure that the ontology was up to date and captured the current versions of the apps at the time when the evaluation was done. This made it necessary to check the ontology to ensure that the permissions requested by the newer versions of the apps continued to work in the same manner and if not, to update the ontology accordingly. So, the ontology was first created with individuals to represent the Ticketmaster app version 1.57.0 and then 1.58.0 for Android, and the Ticketmaster app version 1.76.2 for iPhone. The individuals in the ontology representing the Ticketmaster apps for both the mobile phone with Android and the iPhone mobile phone, were updated to the app version 205.0

for the use case. Even with the updates to the apps, the ontology knowledge of how the settings and permissions interact did not need to be updated.

5.3. Threats to Internal Validity

Here, I discuss the threats to internal validity as well as what I have done to mitigate the threats. I have made efforts to create the SIG, the taxonomy, and the ontology, systematically and logically. To make it easier to navigate the SIGs I have decomposed the softgoals in a systematic manner, as recommended in [44]. So, for example, in Appendix G, the Data Subject Privacy shows Personal Data decomposed by NFR into: Data Subject Control, Ownership, and Rights. I believe that including the topic [Personal Data], makes it clear that, in this case, these subgoals refer specifically to Personal Data. The SIGs were created with information from the literature but as indicated in [34], [43], the technologies continue to be developed and along with them, new possible privacy threats may surface. Therefore, the SIGs will need to be updated, appended to, and maintained to keep them current.

As mentioned previously, the taxonomy is subjective; others may choose different criteria to classify the settings and permissions. Again, as recommended in [55], I classify the settings and permissions based on what they control. Even so, when creating the ontology, I could have created a Setting class and its subclasses, for each operating system. So, for example, I could have created an AndroidSetting class (or a completely separate ontology) for the Android operating system and done the same for the Windows and iOS operating systems. Instead, I chose to create one Setting class with the subclasses lower in the hierarchy distinguished by name for each operating system. So, I have an AndroidDeviceSetting class, a WindowsPrivDevSetting class, and an iOSPrivDevSetting class. Other developers may classify the settings and permissions differently.

I have included the steps to follow when navigating the SIG and the ontology, to ensure that the process could be followed by other researchers.

The mobile phone with Android that was used for the use cases had the Developer Options switched "On" (from a previous project), even so, none of the options were actually selected. Another device with Android was not available. The permissions listed in the App Manager on the device, are the same as those in [74], [75], which correspond to the "Dangerous permissions" in [76]. The Developer Options therefore do not appear to have any impact on the way the device manages the permissions for the Ticketmaster app that was used for the use cases.

5.4. Threats to External Validity

Threats to external validity would occur if the devices in the use cases are not commonly used or if the use case scenarios themselves are not representative of a typical data subject's decisions and actions. In this section, I describe the possible threats to external validity. I discuss how I evaluated these threats and how I ensured that they were not going to affect the end results of the evaluation.

I had a limited number of devices and operating system versions that were available to evaluate PrivateMe. With this limitation, came the possible threat that these devices may not be representative of devices that are currently used by the general population. The main concern with the devices, however, is that the settings and permissions modelled in the taxonomy and ontology are current at the time of the research. The mobile phone with Android is a Samsung SM-A500W. It is running Android version 6.0.1. According to [76], Android 6.0 is the minimum operating system version that offers the data subject an opportunity to select and change the permissions after an app is loaded. Older versions of the operating system require that the settings and

permissions be selected at the time of app installation [76]. Therefore, Android version 6.0.1 is current in the way it manages the settings and permissions. The Lenovo tablet is running Windows 10 Home version 1511, which is past the end of its service period [77]. However, since a Ticketmaster app is not available for Windows, this device is used to access the site via the Chrome browser instead. The browser is loaded as a desktop app, therefore it is not dependent on the Windows operating system device settings and app permissions for access to the device location [73]. The Chrome browser version 83.0.4103.97 was current as at the time of evaluating PrivateMe. Each time the browser version is checked, however, via Help>>About Google Chrome, it automatically updates; the current version is 84.0.4147.125 and is now installed on the tablet. Finally, the iPhone mobile phone is the most up-to-date of the three devices since it was purchased within the last year. The iPhone 8 used for the research is running iOS 13.5.1. I am confident that it provides a good representation of the device settings and app permissions for iOS.

The use case scenarios are intended to be representative of the typical decisions that a data subject would need to make when installing an app on a device and selecting the appropriate settings and permissions. There are three use cases to evaluate PrivateMe. Use case 1 shows how PrivateMe can be used to aid the data subject to match the data that is used by the app to the privacy softgoals, and to select the corresponding privacy settings and permissions (RQ1). Use Case 2 shows how PrivateMe can be used to resolve any conflicts between settings and permissions selected for the app and any other apps on the same device (RQ2). Use Case 3 shows how PrivateMe can be used to ensure that there are no conflicts between the privacy settings and permissions selected on multiple devices, for the apps or the site (RQ3). Although there is one use case per research question, it may be more typical that selecting settings and permissions, and resolving conflicts are not done on distinctly separate occasions. The actual steps that the data

subject follows to use PrivateMe, however, would be the same regardless of whether they are completed on one occasion or not.

A data subject's sense of privacy and willingness to trust is subjective. Therefore, threats to external validity may also arise due to the subjective nature of privacy and trust. To minimize these threats, the data subject in the use cases is neither too private, nor too willing to share personal information. Similarly, the data subject's decision to trust the Ticketmaster service, data controller, and technology, is intended to be reasonable. Use Case 1 includes some examples as to how such a decision can be made, such as reviewing the privacy policy. According to [26], however, most people do not check the privacy policy. Although it may not be possible to mitigate this threat entirely, Use Case 1 shows how PrivateMe can be used to aid the data subject with such decisions.

Apps that offer to find the device if it is lost, rely on having the Location device setting "On". The Location device setting in Use Case 2, however, is switched "On" for the Ticketmaster app to access the location, whilst the Camera permission to access the location is subsequently switched "Off". In the Use Case, the Location device setting was previously "Off". Having the Location device setting "Off" may or may not be the way most people choose to have their setting. The Use Cases are intended to evaluate that PrivateMe can find conflicts between the settings and permissions. Therefore, it is important that the PrivateMe shows the conflicts when a change in the initial state affects the net result of other settings and permissions. Consequently, the starting state of the settings and permissions in the Use Case is not a concern.

5.5. Future Work

This research aims to help the data subject to control private data, according to the privacy goals (RQ1), and to avoid conflicts between the settings and permissions (RQ2), across multiple devices, apps, and sites (RQ3). PrivateMe includes SIGs which offer the data subject a way to understand the privacy threats associated with data use, and to assess the threats against the benefit of using the data and trust. PrivateMe also includes a taxonomy and ontology of privacy settings and permissions which offer the data subject a way to control interactions between the settings and permissions. To further this research, future work may include conducting a study, a survey, or both, with human subjects, to confirm the efficacy of PrivateMe. The discussions for the results in Use Case 2, Use Case 3, and the additional findings, indicate that the knowledge in the ontology for the AppSetting and the AppPermission classes could be developed further. Finally, PrivateMe may be packaged as an app for ease of use, particularly to make it easier to navigate the SIG, the taxonomy, and the ontology. I describe these points in further detail in the following paragraphs.

An empirical study or a survey with human subjects could confirm the benefits of PrivateMe. Such studies could evaluate the use of graphical representations (such as SIGs) as a means to help the data subject to understand and assess privacy goals, which include: the data that is collected by apps, the service or services that are realized by using the data, the possible privacy threats, and trust. Conducting an empirical study over a period of time, could also confirm if packaging PrivateMe as an app would be helpful to the data subject. This may make it easier for them to use PrivateMe interactively as they load apps on their devices, decide which privacy settings and permissions will meet their privacy goals, and resolve any conflicts between the settings and permissions.

Use Case 2 discusses the AppSetting class in the ontology. As future work, the AppSetting class could be developed further to show the difference between an app setting and an app permission. For example, the AppSetting class could be developed to show how on the device with Android, the Camera app setting for Location configures it to tag photos but to do so, it needs app permission to access the location on the device. This also requires that the device Location setting is "On". Although many app settings are specific to the app, the AppSetting class could also be developed further to include examples of some common settings such as Account Profile or History. Along with this, as discussed in Use Case 3, the corresponding SiteSetting class could be developed to show the similarity between the AppSetting and the SiteSetting classes.

Use Case 3 shows that site permissions are managed via the browser. The granularity of the app permissions requested by an app, differ depending on the operating system of the device. The site permissions may similarly differ depending on the browser. The ontology could therefore be developed further to include different browsers to capture how the site permissions are managed via the browser. Use Case 3 was focused on the Location permissions for the app and site. The AppPermission and SitePermission subclasses of the Permission class may also be developed further to include additional permissions such as Body or Health sensors, or the Microphone.

As part of the future work, PrivateMe could be developed into an application (or app) to make it easier for the data subject to navigate the SIGs and the ontology. It could provide a more interactive interface than static SIGs to display the privacy goals and potential privacy threats. Validation testing for a new version of the ontology could be conducted with devices that are current at the time of testing. These devices would ideally be used solely for the research, to ensure that no automatic updates are switched on and that no other atypical features are enabled.

References

[1] "Legal - Copyright and Trademark Guidelines - Apple." https://www.apple.com/legal/intellectual-property/guidelinesfor3rdparties.html (accessed Sep. 18, 2020).
[2] "Microsoft Trademark & Brand Guidelines | Trademarks." https://www.microsoft.com/en-us/legal/intellectualproperty/trademarks/usage/general (accessed Sep. 20, 2020).
[3] "Publications, Seminars, & Conference Guidelines | Trademarks." https://www.microsoft.com/en-us/legal/intellectualproperty/trademarks/usage/publications.aspx (accessed Sep. 20, 2020).
[4] "Permissions – Google." https://www.google.com/permissions/logos-trademarks/ (accessed Sep. 19, 2020).
[5] "Permissions – Google." https://www.google.com/permissions/trademark/brand-terms/ (accessed Sep. 19, 2020).
[6] "Partner Marketing Hub." https://partnermarketinghub.withgoogle.com/#/legal-line-generator (accessed Sep. 19, 2020).
[7] "Legal - Trademark List - Apple." https://www.apple.com/legal/intellectual-property/trademark/appletmlist.html (accessed Sep. 18, 2020).
[8] "EUR-Lex - 02016R0679-20160504 - EN - EUR-Lex." https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A02016R0679-20160504 (accessed Jul. 30, 2018).
[9] "Brand guidelines | Google Play | Android Developers." https://developer.android.com/distribute/marketing-tools/brand-guidelines (accessed Sep. 18, 2020).
[10] "Permissions – Google." https://www.google.com/permissions/ (accessed Sep. 19, 2020).
[11] "Permissions – Google." https://www.google.com/permissions/products/ (accessed Sep. 19, 2020).
[12] "Permissions – Google." https://www.google.com/permissions/faq/ (accessed Sep. 19, 2020).
[13] "Partner Marketing Hub." https://partnermarketinghub.withgoogle.com/#/faq (accessed Sep. 25, 2020).
[14] "Understand locations & edit them - Computer - Google Photos Help." https://support.google.com/photos/answer/6153599 (accessed Sep. 23, 2019).
[15] C. Sengul, "Privacy in the Internet of Things: Regulation vs Innovation," *IEEE Internet of Things*, Sep. 07, 2016. https://iot.ieee.org/newsletter/september-2016/privacy-in-the-internet-of-things-regulation-vs-innovation.html (accessed Jul. 17, 2018).
[16] G. Vojkovic, "Will the GDPR slow down development of smart cities?," pp. 1295–1297, May 2018.
[17] "INFERENCE | meaning in the Cambridge English Dictionary." https://dictionary.cambridge.org/dictionary/english/inference (accessed Oct. 08, 2019).
[18] J. Fernquist, T. Fangstrom, and L. Kaati, "IoT Data Profiles: The Routines of Your Life Reveals Who You Are," in *2017 European Intelligence and Security Informatics Conference (EISIC)*, Athens, Sep. 2017, pp. 61–67, doi: 10.1109/EISIC.2017.17.
[19] J. A. Colley and A. Crabtree, "Object Based Media, the IoT and Databox," in *Living in the Internet of Things: Cybersecurity of the IoT - 2018*, London, UK, 2018, p. 34 (6 pp.)-34 (6 pp.), doi: 10.1049/cp.2018.0034.
[20] E. Krempel and J. Beyerer, "The EU General Data Protection Regulation and its Effects on Designing Assistive Environments," in *Proceedings of the 11th PErvasive Technologies Related to Assistive Environments Conference on - PETRA '18*, Corfu, Greece, 2018, pp. 327–330, doi: 10.1145/3197768.3201567.
[21] Sz. Varadi, G. G. Varkonyi, and A. Kertesz, "Law and IoT: How to see things clearly in the Fog," in *2018 Third International Conference on Fog and Mobile Edge Computing (FMEC)*, Barcelona, Apr. 2018, pp. 233–238, doi: 10.1109/FMEC.2018.8364070.
[22] "Rights for citizens," *European Commission - European Commission*. https://ec.europa.eu/info/law/law-topic/data-protection/reform/rights-citizens_en (accessed Mar. 06, 2019).

[23] I. Omoronyia, L. Cavallaro, M. Salehie, L. Pasquale, and B. Nuseibeh, “Engineering adaptive privacy: On the role of privacy awareness requirements,” in *2013 35th International Conference on Software Engineering (ICSE)*, San Francisco, CA, USA, May 2013, pp. 632–641, doi: 10.1109/ICSE.2013.6606609.
[24] I. Omoronyia, L. Pasquale, M. Salehie, L. Cavallaro, G. Doherty, and B. Nuseibeh, “Caprice: a tool for engineering adaptive privacy,” in *Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering - ASE 2012*, Essen, Germany, 2012, p. 354, doi: 10.1145/2351676.2351745.
[25] P. Anthonysamy, P. Greenwood, and A. Rashid, “Social Networking Privacy: Understanding the Disconnect from Policy to Controls,” *Computer*, vol. 46, no. 6, pp. 60–67, Jun. 2013, doi: 10.1109/MC.2012.326.
[26] S. Zimmeck *et al.*, “Automated Analysis of Privacy Requirements for Mobile Apps,” presented at the AAAI Fall Symposium Series, North America, Sep. 2016, Accessed: Aug. 27, 2019. [Online]. Available: https://www.aaai.org/ocs/index.php/FSS/FSS16/paper/view/14113/13704.
[27] H. Almuhimedi *et al.*, “Your Location has been Shared 5,398 Times!: A Field Study on Mobile App Privacy Nudging,” in *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI ’15*, Seoul, Republic of Korea, 2015, pp. 787–796, doi: 10.1145/2702123.2702210.
[28] J. T. Frece, “The challenge of OwnData service features: A step towards an informed choice of an OwnData service,” in *2017 Global Internet of Things Summit (GIoTS)*, Geneva, Switzerland, Jun. 2017, pp. 1–6, doi: 10.1109/GIOTS.2017.8016283.
[29] T. T. Tun *et al.*, “Privacy arguments: Analysing selective disclosure requirements for mobile applications,” in *2012 20th IEEE International Requirements Engineering Conference (RE)*, Chicago, IL, USA, Sep. 2012, pp. 131–140, doi: 10.1109/RE.2012.6345797.
[30] M. Langheinrich, “A Privacy Awareness System for Ubiquitous Computing Environments,” in *UbiComp 2002: Ubiquitous Computing*, vol. 2498, G. Borriello and L. E. Holmquist, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 237–245.
[31] F. Schaub, B. Konings, M. Weber, and F. Kargl, “Towards context adaptive privacy decisions in ubiquitous computing,” in *2012 IEEE International Conference on Pervasive Computing and Communications Workshops*, Lugano, Switzerland, Mar. 2012, pp. 407–410, doi: 10.1109/PerComW.2012.6197521.
[32] “PRIVACY | meaning in the Cambridge English Dictionary.” https://dictionary.cambridge.org/dictionary/english/privacy (accessed Jun. 07, 2019).
[33] “Definition of PRIVACY.” https://www.merriam-webster.com/dictionary/privacy (accessed Jun. 07, 2019).
[34] D. J. Solove, *Understanding privacy*. Cambridge, Mass: Harvard University Press, 2008.
[35] L. Palen and P. Dourish, “Unpacking ‘Privacy’ for a Networked World,” in *NEW HORIZONS*, 2003, vol. 5, pp. 129–136, doi: http://dx.doi.org/10.1145/642611.64263.
[36] A. S. Ahmadian, J. Jürjens, and D. Strüber, “Extending model-based privacy analysis for the industrial data space by exploiting privacy level agreements,” in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing - SAC ’18*, Pau, France, 2018, pp. 1142–1149, doi: 10.1145/3167132.3167256.
[37] “THREAT | meaning in the Cambridge English Dictionary.” https://dictionary.cambridge.org/dictionary/english/threat (accessed Jun. 19, 2020).
[38] R. Veleda and L. M. Cysneiros, “Towards a Tool to Help Exploring Existing Non-functional Requirements Solution Patterns,” in *2017 IEEE 25th International Requirements Engineering Conference Workshops (REW)*, Lisbon, Portugal, Sep. 2017, pp. 232–239, doi: 10.1109/REW.2017.49.
[39] O. Zinovatna and L. M. Cysneiros, “Reusing knowledge on delivering privacy and transparency together,” in *2015 IEEE Fifth International Workshop on Requirements Patterns (RePa)*, Ottawa, ON, Canada, Aug. 2015, pp. 17–24, doi: 10.1109/RePa.2015.7407733.

[40] C. Kalloniatis, E. Kavakli, and S. Gritzalis, “Addressing privacy requirements in system design: the PriS method,” *Requirements Engineering*, vol. 13, no. 3, pp. 241–255, Sep. 2008, doi: 10.1007/s00766-008-0067-3.
[41] L. Liu, E. Yu, and J. Mylopoulos, “Security and privacy requirements analysis within a social setting,” presented at the 11th IEEE International Requirements Engineering Conference, 2003, Accessed: Aug. 29, 2018. [Online].
[42] A. S. Ahmadian, D. Strüber, V. Riediger, and J. Jürjens, “Supporting privacy impact assessment by model-based privacy analysis,” in *Proceedings of the 33rd Annual ACM Symposium on Applied Computing - SAC ’18*, Pau, France, 2018, pp. 1467–1474, doi: 10.1145/3167132.3167288.
[43] E.-L. Makri and C. Lambrinoudakis, “Privacy Principles: Towards a Common Privacy Audit Methodology,” in *Trust, Privacy and Security in Digital Business*, vol. 9264, S. Fischer-Hübner, C. Lambrinoudakis, and J. López, Eds. Cham: Springer International Publishing, 2015, pp. 219–234.
[44] L. Chung, B. A. Nixon, E. Yu, and J. Mylopoulos, *Non-Functional Requirements in Software Engineering*. Norwell, MA: Kluwer Academic Publishers, 2000.
[45] J. W. Satzinger, R. B. Jackson, and S. D. Burd, *Systems Analysis and Design in a Changing World*, 6th ed. Boston, MA: Course Technology, 2012.
[46] B. Bruegge and A. H. Dutoit, *Object-Oriented Software Engineering Using UML, Patterns, and Java*, 3rd ed. Boston: Pearson, 2010.
[47] N. Rozanski and E. Woods, *Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives*, 2nd ed. New Jersey: Pearson Education, Inc., 2012.
[48] L.M. Cysneiros, private communication, Feb. 27, 2019.
[49] S. Sapukkal and L. Chung, “RE-Tools: A Multi-notational Requirements Modeling Toolkit.” (3.0.2). Accessed: Sep. 05, 2018. [Online]. Available: https://personal.utdallas.edu/~supakkul/tools/RE-Tools/index.html
[50] StarUML. (5.0.2.1570). Accessed Sep. 05, 2018. [Online]. Available: https://sourceforge.net/projects/staruml/
[51] J. Larus *et al.*, “When Computers Decide: European Recommendations on Machine-Learned Automated Decision Making,” ACM, Jan. 2018. doi: 10.1145/3185595.
[52] C. Huber, M. Chaudhary, M. Shtern, J. Mukherjee, M. Litoiu, and V. Onut, “Privacy-aware efficient visual recognition services for smart laboratories.” Unpublished, Oct. 30, 2018.
[53] “TAXONOMY | meaning in the Cambridge English Dictionary.” https://dictionary.cambridge.org/dictionary/english/taxonomy.
[54] M.A. Musen, “The Protege project: A look back and a look forward.,” *AI Matters*, vol. 1, no. 4, Jun. 2015, doi: 10.1145/2557001.25757003.
[55] N. F. Noy and D. L. McGuinness, “Ontology Development 101: A Guide to Creating Your First Ontology.” Stanford University, Stanford, CA, Accessed: Jan. 16, 2019. [Online]. Available: https://protege.stanford.edu/publications/ontology_development/ontology101.pdf.
[56] A. Rector, “Protege4Pizzas10Minutes - Protege Wiki,” May 23, 2016. https://protegewiki.stanford.edu/wiki/Protege4Pizzas10Minutes (accessed Dec. 11, 2018).
[57] M. Horridge, “A Practical Guide To Building OWL Ontologies Using Prot´eg´e 4 and CO-ODE Tools. Edition 1.2.” The University of Manchester, Mar. 13, 2009, Accessed: Jan. 28, 2019. [Online]. Available: http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_2.pdf.
[58] “INTRINSIC | meaning in the Cambridge English Dictionary.” https://dictionary.cambridge.org/dictionary/english/intrinsic (accessed Feb. 07, 2020).
[59] W3C OWL Working Group, Ed., “OWL2 Web Ontology Language Document Overview (Second Edition).” W3C (MIT, ERCIM, Keio), 2012, Accessed: Feb. 07, 2019. [Online]. Available: https://www.w3.org/TR/2012/REC-owl2-overview-20121211/.
[60] Pascal Hitzler, Markus Krotzsch, Peter F. Patel-Schneider, and Sebastian Rudolph, Eds., “OWL2 Web Ontology Language Primer (Second Edition).” 2012 W3c (MIT, ERCIM, Keio), Accessed: Feb. 08, 2019. [Online]. Available: https://www.w3.org/TR/2012/REC-owl2-primer-20121211/.

[61] M. Horridge, "A Practical Guide To Building OWL Ontologies Using Prot´eg´e 4 and CO-ODE Tools Edition 1.3." The University of Manchester, Mar. 24, 2011, Accessed: Feb. 23, 2019. [Online]. Available: http://mowl-power.cs.man.ac.uk/protegeowltutorial/resources/ProtegeOWLTutorialP4_v1_3.pdf.
[62] G. Schreiber, "OWL restrictions." May 31, 2005, Accessed: Feb. 12, 2019. [Online]. Available: https://www.cs.vu.nl/~guus/public/owl-restrictions/.
[63] "Definition of SEMANTIC." https://www.merriam-webster.com/dictionary/semantic (accessed Mar. 05, 2020).
[64] "SEMANTIC WEB | meaning in the Cambridge English Dictionary." https://dictionary.cambridge.org/dictionary/english/semantic-web (accessed Mar. 05, 2020).
[65] "Permissions – Google." https://www.google.com/permissions/trademark/rules/ (accessed Sep. 19, 2020).
[66] J. P. McCrae and C. Unger, "Design Patterns for Engineering the Ontology-Lexicon Interface," in *Towards the Multilingual Semantic Web*, P. Buitelaar and P. Cimiano, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014, pp. 15–30.
[67] M. Uschold, "Demystifying OWL for the Enterprise," *Synthesis Lectures on the Semantic Web: Theory and Technology*, vol. 8, no. 1, pp. i–237, May 2018, doi: 10.2200/S00824ED1V01Y201801WBE017.
[68] "Permissions – Google." https://www.google.com/permissions/trademark/trademark-list/ (accessed Sep. 19, 2020).
[69] "Partner Marketing Hub." https://partnermarketinghub.withgoogle.com/#/brands/1KdZxnHbtjVwReGOY03iKT8DDx1Y34joI/1AE5SD4G3_SnLn4KNo8BzHLVsHz-JvNauEEVpRYlNm_Q (accessed Sep. 19, 2020).
[70] "About Ticketmaster, a Live Nation Entertainment, Inc company." https://www.ticketmaster.ca/about/about-us.html (accessed May 11, 2020).
[71] "Ticketmaster Privacy Policy." https://help.ticketmaster.ca/s/article/Ticketmaster-Privacy-Policy?language=en_US (accessed Apr. 03, 2020).
[72] "Windows 10 location service and privacy – Microsoft privacy." https://support.microsoft.com/en-us/help/4468240/windows-10-location-service-and-privacy (accessed Dec. 03, 2019).
[73] "Windows 10 desktop apps and privacy." https://support.microsoft.com/en-us/help/4468234/windows-10-desktop-apps-and-privacy (accessed Oct. 02, 2019).
[74] "Manifest.permission_group | Android Developers." https://developer.android.com/reference/android/Manifest.permission_group (accessed Jul. 19, 2020).
[75] B. Poiesz. *Google I/O 2015 - Android M Permissions*. (2015). Accessed: Jul. 17, 2020. [Online Video]. Available: https://www.youtube.com/watch?v=f17qe9vZ8RM
[76] "Permissions overview," *Android Developers*. https://developer.android.com/guide/topics/permissions/overview (accessed Jul. 17, 2020).
[77] "Windows lifecycle fact sheet - Windows Help." https://support.microsoft.com/en-us/help/13853/windows-lifecycle-fact-sheet (accessed May 01, 2020).
[78] M. Boban and M. Weber, "Internet of Things, legal and regulatory framework in digital transformation from smart to intelligent cities," Opatija Croatia, May 2018, pp. 1359–1364, Accessed: Jul. 16, 2018. [Online].

Appendices

Appendix A. Privacy Concerns

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Ferquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Tun et al., 2012
Concern of/with	**About**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[29]
Automated Decision Making	Bias		x												
Automated Decision Making	Explainable Decision Making					x									
Data	Aggregation							x							
Data	Confidentiality	x													
Data	Data Integrity			x											
Data	Data Minimization								x						
Data	Data Persistence		x												
Data	Disclosure														x
Data	Future Use							x							
Data	Indirect Data Collection				x										x
Data	IoT User Identification		x					x							
Data	User Activities		x					x							
Data	User Location		x	x			x	x						x	
Data	User Profiles		x					x							
Data	Volume	x	x											x	x

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Ferquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Tun et al., 2012
Concern of/with	**About**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[29]
Data Controller	Interests		x												
Data Controller	Market Share	x													
Data Controller	Rights	x													
Data Subject	Control of Personal Data		x	x			x								
Data Subject	Data Ownership	x													
Data Subject	Freedom				x										
Data Subject	Personal safety							x							
Data Subject	Privacy	x		x			x	x				x	x		x
Data Subject	Rights, including Informed Consent		x	x		x						x			
Data Subject	Unaware of data collection/use							x				x			
Ethics	Obtaining Consent			x											
Ethics	Ethical Use	x					x								
Machine Learning	Bias in Algorithm					x									
Machine Learning	Bias in Training Data					x									
Machine Learning	Confusing the model					x									
Machine Learning	Effective, feasible testing					x									
Machine Learning	Social Engineering					x									
Privacy	Complexity													x	
Privacy	Context Awareness												x	x	x

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Ferquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Tun et al., 2012
Concern of/with	**About**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[29]
Privacy	Social Engineering					x									
Privacy	User-Managed Privacy									x	x				
Security	Data Breach				x										
Security	Data Protection - Org				x										
Security	Data Protection - Tech				x										
Security	Data Security	x						x							
Security	Data Transfer			x											
Security	IoT Architecture	x													
Security	Privacy Breach				x										
Security	Standards	x													
Service Provider	Accountability						x								
Service Provider	Legal Compliance						x								
Trust	Trusted 3rd Party								x						
Trust	User trust in Service			x			x		x						
Trust	User trust in Technology	x					x								

Appendix B. Concerns and Corresponding Threats

Concerns that are not related to Privacy Threats:

- Data Controller interests, market share, rights
- Trust

Missing from the list:

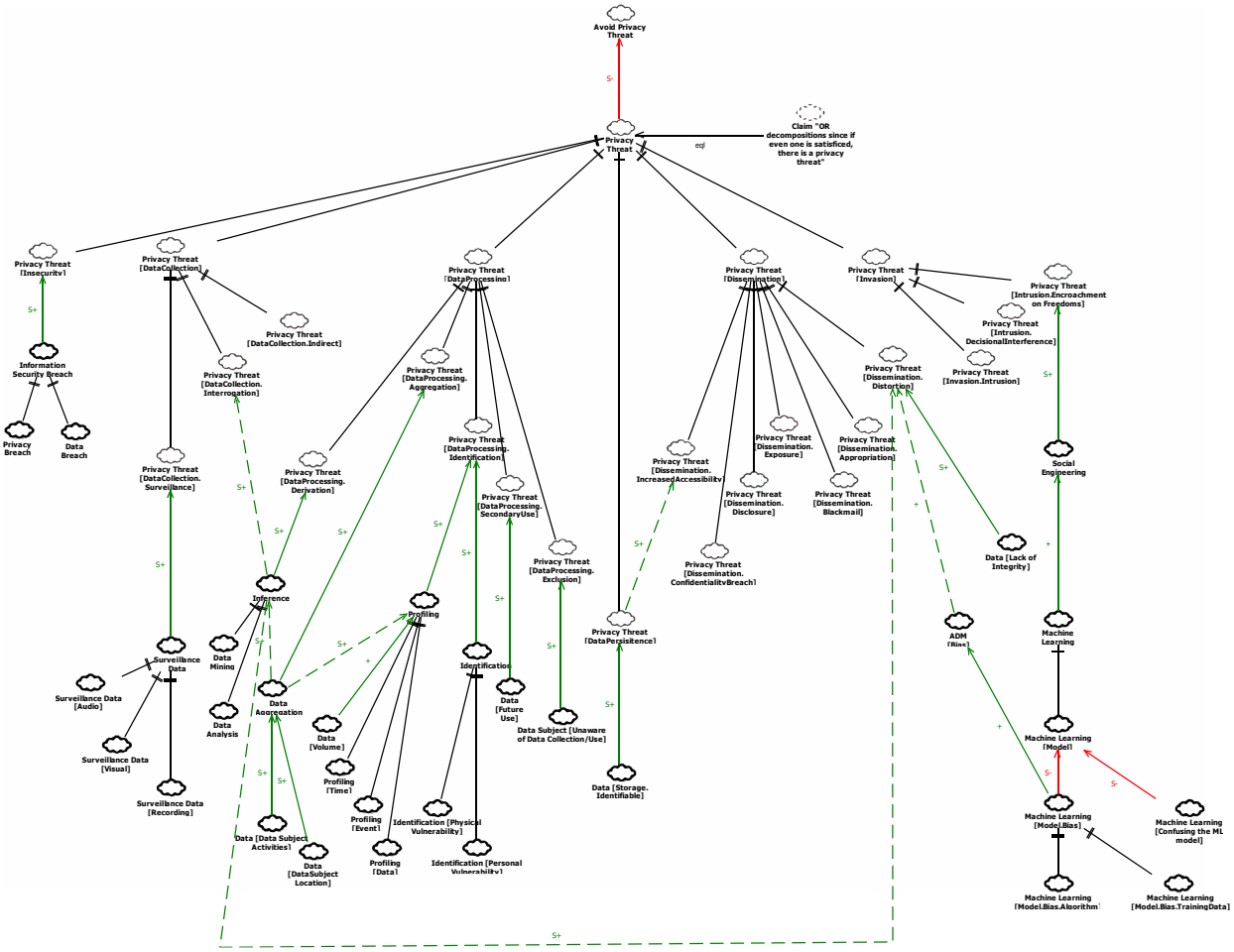
- Inference (i.e. determining data values from other pieces of data [23], as opposed to indirect data collection [15] , which may be unintentional)

Concern of/with	To Prevent Privacy Threat	To Protect from Privacy Threat	Direct Privacy Threat	Privacy Threats [34]
Automated Decision Making			Bias	Distortion
Automated Decision Making	Explainable Decision Making			Distortion
Data			Aggregation	Aggregation
Data	Confidentiality			Breach of Confidentiality
Data	Data Integrity			Distortion
Data	Data Minimization			Increased Accessibility, Secondary Use
Data			Data Persistence	Increased Accessibility, Secondary Use
Data			Disclosure	Disclosure
Data			Future Use	Secondary Use
Data			Indirect Data Collection	Exclusion
Data			IoT User Identification	Identification
Data			User Activities	Surveillance
Data			User Location	Surveillance
Data			**User Profiles**	**Identification**

Concern of/with	To Prevent Privacy Threat	To Protect from Privacy Threat	Direct Privacy Threat	Privacy Threats [34]
Data			Volume	Identification, Increased Accessibility
Data Subject	Control of Personal Data			
Data Subject	Data Ownership			
Data Subject		Freedom		Surveillance, Interrogation, Intrusion
Data Subject		Personal safety		Intrusion
Data Subject		Privacy		All
Data Subject	Rights, including Informed Consent			Exclusion
Data Subject			Unaware of data collection/use	Exclusion
Ethics	Obtaining Consent			Exclusion
Ethics	Ethical Use			Secondary Use, Exclusion, Insecurity
Machine Learning			Bias in Algorithm	Distortion
Machine Learning			Bias in Training Data	Distortion
Machine Learning			Confusing the model	Distortion
Machine Learning	Effective, feasible testing			Distortion
Machine Learning			Social Engineering	
Privacy			Complexity	Multiple
Privacy	Context Awareness			Disclosure
Privacy			Social Engineering	Decisional Interference
Privacy	User-Managed Privacy			N/A
Security			Data Breach	Dissemination
Security	Data Protection Organizational measures			Insecurity, Breach of Confidentiality, Disclosure

Concern of/with	To Prevent Privacy Threat	To Protect from Privacy Threat	Direct Privacy Threat	Privacy Threats [34]
Security	Data Protection Technical measures			Insecurity, Breach of Confidentiality, Disclosure
Security	Data Security			Insecurity, Breach of Confidentiality, Disclosure
Security	Data Transfer			Insecurity, Breach of Confidentiality, Disclosure
Security	IoT Architecture			Insecurity, Breach of Confidentiality, Disclosure
Security			Privacy Breach	Any
Security	Standards			Insecurity, Breach of Confidentiality, Disclosure
Service Provider	Accountability			Multiple
Service Provider	Legal Compliance			Multiple

Appendix C. Privacy Threats Softgoal Interdependency Graph (SIG)



Appendix D. General Concerns

(Note that [23] uses inference for their model, therefore it is not listed as a concern, here.)

Legend

- Privacy Threat
- Prevents privacy threat (operationalization)
- Business Concerns/ Data Controller and Data Subject shared concerns
- To be protected from privacy threats
- Social Concerns

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Automated Decision Making	Bias		x			x										
Automated Decision Making	Explainable Decision Making					x										

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Automated Decision Making	Model Accuracy								x							
Automated Decision Making	Reliability					x										
Business	Agility	x														
Business	Customer focus	x														
Business	Customer Retention								x							
Business	Decision Support					x										
Business	Economy	x				x										
Business	Efficiency	x		x	x	x										
Business	Globalization	x														
Business	Innovation	x													x	
Business	Market Share	x														
Business	Process automation					x										

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Business	Profitability (Business Interests)		x			x										
Business	Resource sharing			x												
Business	Service offered	x				x	x									
Business	Strategy	x														
Business	Technical Feasibility								x		x				x	
Data	Aggregation	x						x							x	
Data	Confidentiality	x			x										x	
Data	Data Integrity	x		x											x	
Data	Data Minimization		x	x	x				x						x	
Data	Data Mining														x	
Data	Data Persistence		x												x	
Data	Disclosure															x

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Data	Indirect Data Collection				x										x	
Data	IoT User Identification		x					x							x	
Data	Secondary / Future Use		x					x	x		x					
Data	**User Activities**		x					x						x		
Data	**User Location**		x	x			x	x						x		
Data	**User Profiles**		x					x								
Data	Volume	x	x						x					x		x
Data Controller	Reputation			x												
Data Controller	Rights	x	x	x	x											
Data Controller	Transparency		x	x	x	x									x	
Data Subject	Control of Personal Data	x	x	x			x								x	
Data Subject	Data Ownership	x	x												x	

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Data Subject	Personal Freedom				x			x								
Data Subject	Personal Safety							x								
Data Subject	Privacy	x		x			x	x	x		x	x	x		x	x
Data Subject	Quality of life	x			x			x								
Data Subject	Rights, including Informed Consent, easy to use privacy (usability), portability etc		x	x		x			x		x				x	
Data Subject	**Unaware of data collection/use**							x		x	x	x		x		
Ethics	Ethical Use	x					x		x							
Ethics	Obtaining Consent		x	x	x											

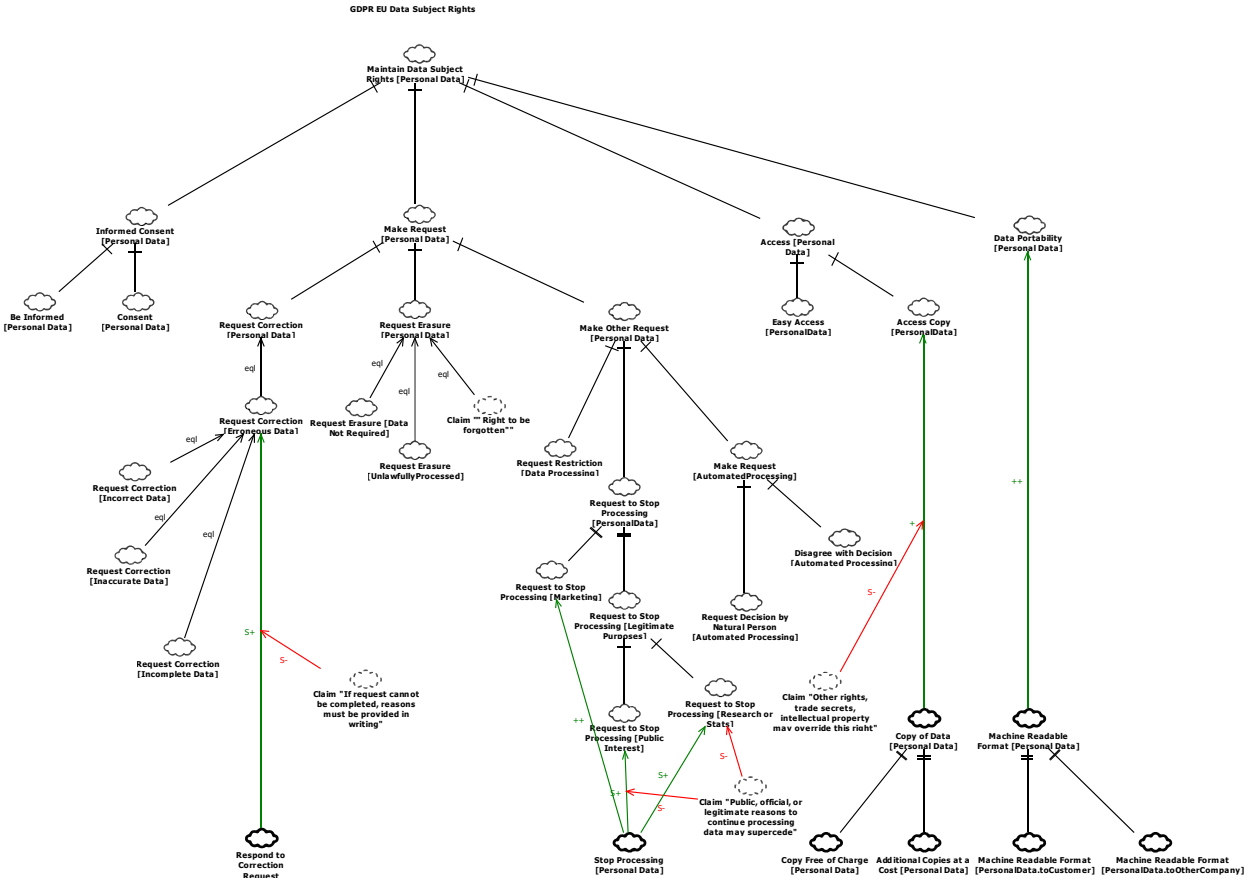
		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Legal	Accountability			x		x	x				x					
Legal	Compliance	x	x	x			x								x	
Legal	Liability			x												
Legal	Responsibility			x		x										
Legal	Traceability					x										
Machine Learning	Bias in Algorithm					x										
Machine Learning	Bias in Training Data					x										
Machine Learning	Confusing the model					x										
Machine Learning	Effective, feasible testing					x										
Machine Learning	Social Engineering					x										
Privacy	Complexity													x		
Privacy	Context Awareness											x	x	x	x	x

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Privacy	User-Managed Privacy								x	x	x				x	
Information Security	Data Breach				x											
Information Security	Data Collection			x												
Information Security	Data Protection - Org				x											
Information Security	Data Protection - Tech	x	x		x											
Information Security	Data Security			x			x				x				x	
Information Security	Data Storage			x												
Information Security	Data Transfer	x		x							x					
Information Security	IoT Architecture	x		x												

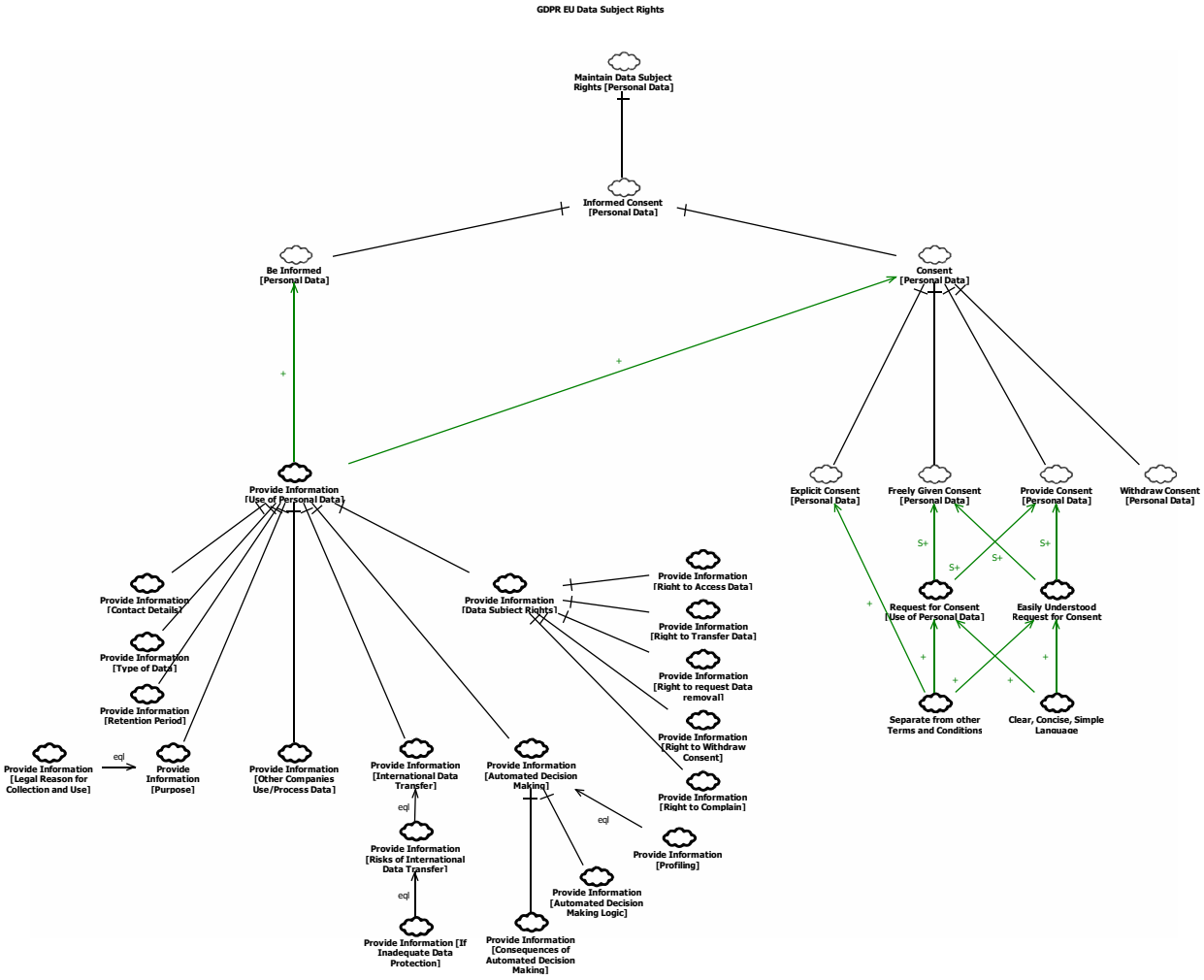
		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Information Security	Privacy Breach				x											
Information Security	Standards	x														
Social	Critical Thinking					x										
Social	Culture	x				x										
Social	Greater good					x										
Social/Economic	Job redundancy					x										
Social/Stewardship	Environment	x														
Social/Stewardship	Sustainability	x														
Software	Customization						x		x		x					
Trust	Data controller trust data subject								x							
Trust	Data subject trust data controller								x							

		Boban and Weber, 2018	Krempel and Beyerer, 2018	Varadi, Varkonyi, and Kertez, 2018	Vojkovic, 2018	Larus et al., 2018	Colley and Crabtree, 2018	Fernquist, Fangstrom, and Kaati, 2017	Frece, 2017	Almuhimedi et al., 2015	Langheinrich, 2002	Omoronyia et al., 2013	Omoronyia et al., 2012	Schaub, Konings, Weber, and Kargl, 2012	Sengul, 2016	Tun et al., 2012
Type of Concern (Concern With. . .)	**Concern (About. . .)**	[78]	[20]	[21]	[16]	[51]	[19]	[18]	[28]	[27]	[30]	[23]	[24]	[31]	[15]	[29]
Trust	Data Subject trust in Service			x			x		x							
Trust	Data Subject trust in Technology	x				x	x							x	x	
Trust	Trusted 3rd Party								x							

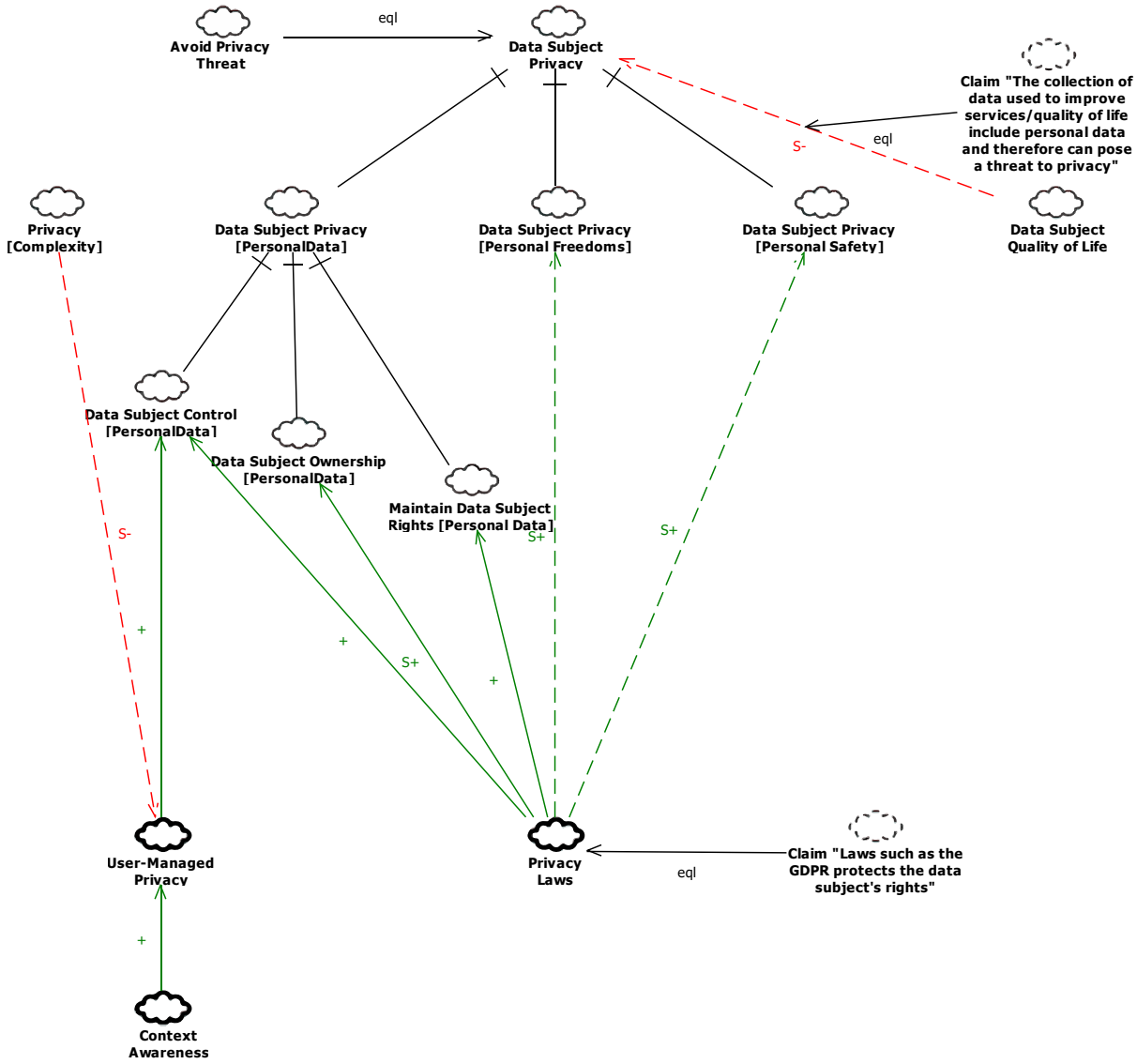
Appendix E. (GDPR) Data Subject Rights [8], [22] SIG



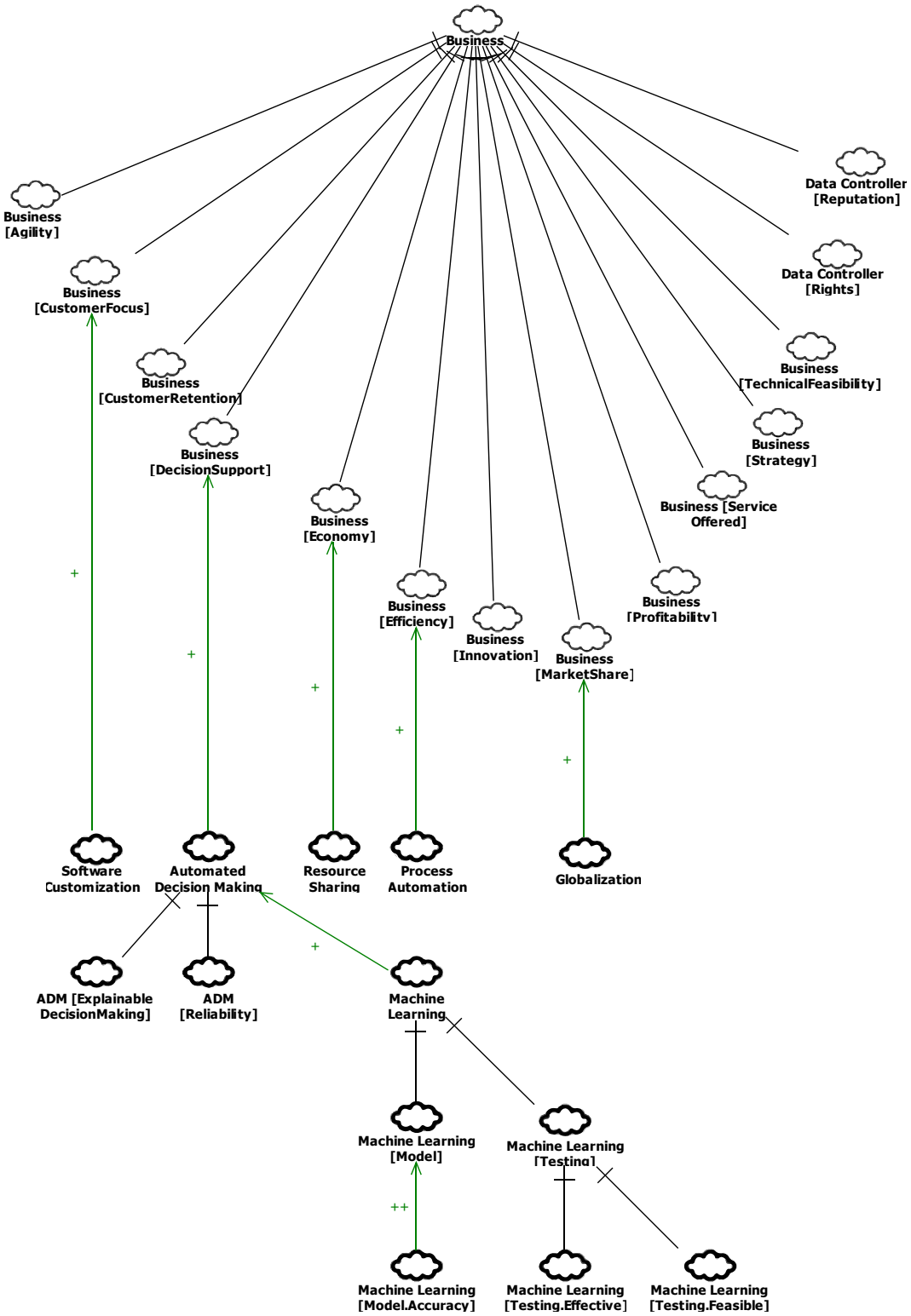
Appendix F. (GDPR) Data Subject Rights [8], [22] (Consent and to Be Informed) SIG



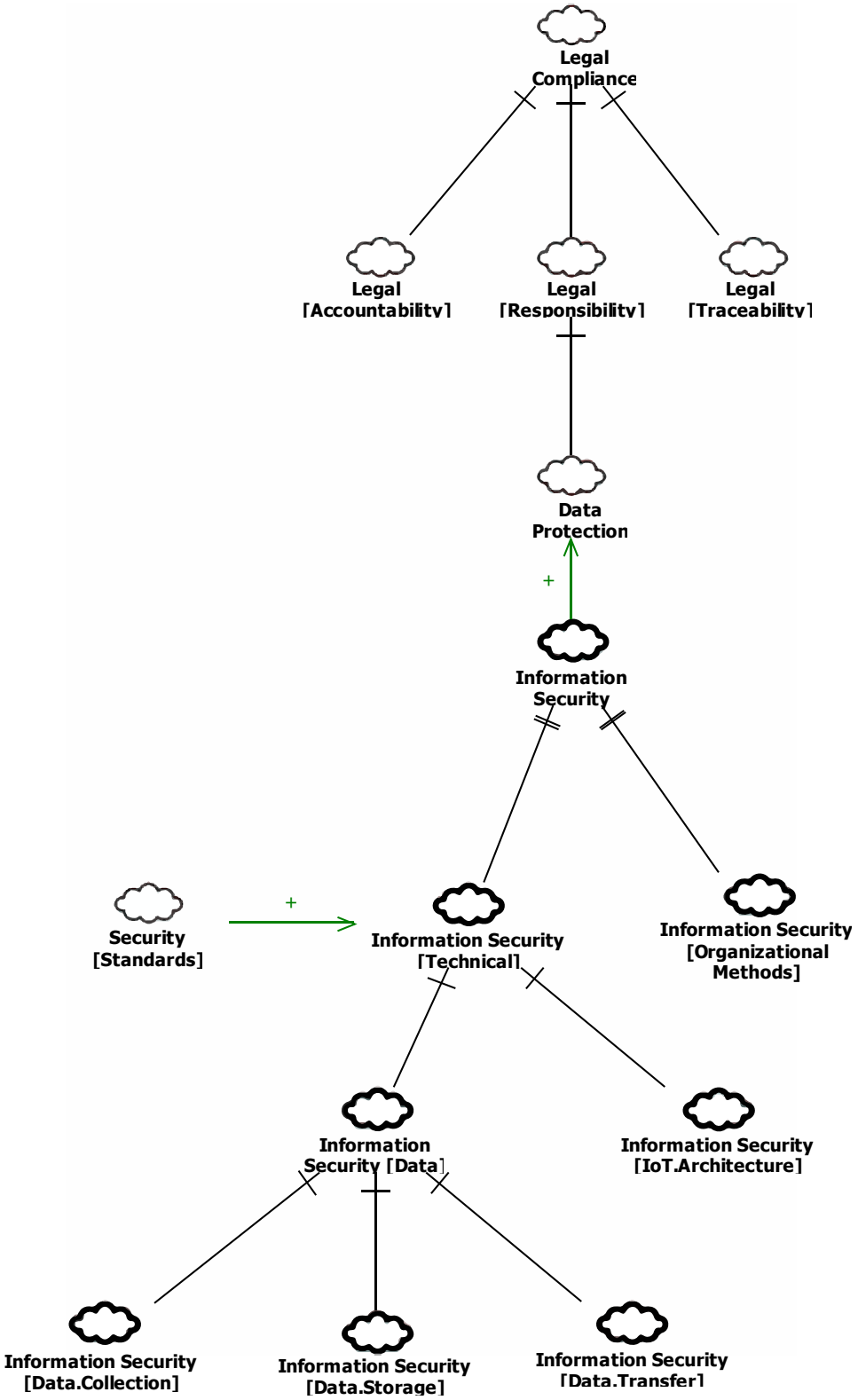
Appendix G. Data Subject Privacy Concerns SIG



Appendix H. Business Concerns SIG



Appendix I. Legal Concerns SIG



Appendix J. Trust SIG