

Designing Privacy-by-Design

Jeroen van Rest, Daniel Boonstra, Maarten Everts,
Martin van Rijn, and Ron van Paassen

TNO, Delft / The Hague, The Netherlands
{Jeroen.vanrest,Daniel.boonstra,Maarten.everts,
Martin.vanRijn,Ron.vanPaassen}@tno.nl

Abstract. The proposal for a new privacy regulation d.d. January 25th 2012 introduces sanctions of up to 2% of the annual turnover of enterprises. This elevates the importance of mitigation of privacy risks. This paper makes Privacy by Design more concrete, and positions it as the mechanism to mitigate these privacy risks.

In this vision paper, we describe how design patterns may be used to make the principle of Privacy by Design specific for relevant application domains. We identify a number of privacy design patterns as examples and we argue that the art is in finding the right level of abstraction to describe a privacy design pattern: the level where the data holder, data subject and privacy risks are described.

We give an extended definition of Privacy by Design and, taking Solove's model for privacy invasions as structuring principle, we describe a tool and method to use that tool to generate trust in systems by citizens.

Keywords: privacy, privacy design pattern, privacy-by-design, system engineering, trust, tooling.

1 Introduction

The European Commission is preparing a reform of the current European data protection directive [1]. In their proposal for a new Data Protection Regulation [2], the Commission stated that it will

"... examine [...] the possibilities for the concrete implementation of the concept of 'privacy by design' [...] to enhance data controllers' responsibility." [3]

An explanation of the principle of Privacy by Design (PbD) is given in a footnote:

"The principle of 'Privacy by Design' means that privacy and data protection are embedded throughout the entire life cycle of technologies, from the early design stage to their deployment, use and ultimate disposal. This principle features inter alia in the Commission Communication on 'A Digital Agenda for Europe' - COM(2010) 245."

The Digital Agenda for Europe embraces the same definition of Privacy by Design with no further explanation or reference [4]. The trail for defining the concept stops there.

Legislation makes extensive references to “privacy by design”, but fails to specify what it means exactly, as has also been pointed out by Van Lieshout [5], who argues that a holistic approach to Privacy by Design indeed offers surplus value but that actual implementation is confronted with difficulties such as lack of economic incentives, legacy systems, and lack of adoption of trust of end-users and consumers in PbD.

The omission of a clear definition of PbD entails that for European citizens, policy makers, authorities and industry it is currently unclear what a request for PbD practically means. Still, many are confronted with such requests, not only in communications from the Commission, but also by local authorities¹ and in calls for the 7th Framework Research Programme. PbD currently is only an apparent solution, not a real one.

1.1 Privacy by Design: History of a Vague Concept

Although the term Privacy by Design was not yet used, a joint paper by the Dutch Data Protection Authority and TNO FEL² is one of the first papers that take a generic look at privacy from a designer’s perspective [7]. The first explicit public reference to the term Privacy by Design can be found in the title of a workshop held at the conference “Computers, Freedom & Privacy 2000: Workshop on Freedom and Privacy by design” [8]. Around the same time, the EU FP5 project PISA focussed on Privacy Incorporated Software Agents under the name of PbD and Privacy Enhancing Technologies [9]. In North America, PbD was cultivated mainly by Cavoukian [10]. Cavoukian stated that PbD is based on seven “foundational principles” [11], but recently other principles, such as *data minimisation* are also emerging [12]. The translation of these principles to actual designs of systems is done by example. Therefore, everybody is free to postulate a particular design (process) as “Privacy by Design”, and we see companies as Microsoft and IBM doing exactly that [13, 14].

1.2 Roles and Responsibilities

Cavoukian hypothesises an evolutionary approach [15] where industry and consumers find out together what works and what not. The advantage of this approach is that it leaves room for innovation whereas a purely government regulated approach would merely leave room to comply. However, a disadvantage of a pure evolutionary approach is the individual nature of it. If each (commercial) party can decide for itself what PbD means in their application domain, then citizens, consumers, local authorities, end users and policy makers will have to understand the differences between how

¹ Charter for a Democratic Use of Video Surveillance [6]: “*It is important to include in these protocols the “Privacy by design” method, which encourages personal data protection to be considered at the early stages of the system design.*” No further explanation of the concept is given.

² With support from the Information & Privacy Commissioner of Ontario of that time, Tom Wright.

each different party implements PbD. This asks a lot of those citizens, end users, etc., and is therefore at least a missed opportunity for transparency with regard to respecting citizen's rights, but also for market transparency.

In every market domain also operational governmental organisations are active in the role of data holder and designer, collecting data and designing and building information systems such as police, defence, healthcare institutes, education, etc. Whether these parts of government also can afford to make the mistakes³ that are inherent to an evolutionary approach remains to be seen, because liability and responsibility may weigh differently on public services than they do on commercial businesses. This is a second disadvantage of a pure evolutionary approach.

An open question is what citizens and consumers actually need to know about PbD in order for the concept to be useful. Perhaps the concept of PbD has the biggest value between industry and government, not between industry and consumers?

PbD emphasises the role of the designer and integrator in preventing privacy breaches. This does not discharge a owner from taking his responsibility, as, in EU laws, the owner of the system (which makes him the data controller/holder) is responsible, not the party that designed or built it. These parties are not even "known" to data protection laws. The designer does have an influence on the use of technology because technology is not neutral as has been stated in the first of Kranzberg's truisms [16]. It has a function, which through its form –its design- is communicated to its users.

Privacy certification (PC) [17] and privacy impact assessment (PIA) are in itself not PbD. One could argue that a system that was designed according to the principles of PbD should have a good PIA result, and should very quickly and easily be certified by a PC.

1.3 (Behavioural) Economic Perspective

If individual and collective privacy interests could be aligned with economic interests then our economic interests would also stimulate privacy. However, a study on the benefits of Privacy Enhancing Technologies [18] found that individual citizens in general do not flock to products and services that protect their privacy better. This should be investigated further, but this first study undermines the potential market mechanism. Borking theorizes [19] that the application of models for customer adoption [20] should be further studied to address this issue.

If we cannot leave it to behavioural economics or a strict evolutionary approach to protect data subjects' privacy, then we may need a joint approach with industry and government together. Already, the proposal for a (new) EU privacy regulation [2] includes sanctions of up to 2% of the annual turnover (in case of international enterprises) or 1.000.000 EUR. Whether this will actually have an impact remains to be seen, but it creates the first economic incentive on an organisational level to address privacy risks. As legislation does not specify *how to mitigate* those risks, that void could be filled by making Privacy by Design more concrete.

³Bad evolutionary changes die off because the phenotypes that carry them are being "punished" by their environment. Who punishes bad forms of PbD? Will this happen before or after people's privacy has been breached?

1.4 Problem Statement

The specific meaning in a particular application domain of the principle of 'Privacy by Design' is unknown. It is an open question what citizens and consumers actually need to know about PbD in order for the concept to "work".

This means that currently PbD as a concept is not usable to communicate trust (or a lack of trust) in particular systems to European citizens and end users. This is a problem for policy-makers, end users, system integrators, system designers, researchers and, last but not least, for citizens themselves, because it keeps alive a sense of opaqueness and general distrust with regard to information systems⁴.

There are no collective resources available, other than by example and/or by industry, as to what Privacy by Design actually means in a particular application domain. Nor is it known what relevant best practices are, what their consequences are, or which methodologies and tools are available. Innovation in protecting privacy is also hampered, because it is unclear to policy makers, citizens and purchasers how to compare innovations (Privacy Enhancing Technologies, PET) on their ability to mitigate privacy risks.

This leaves some questions, which we will attempt to address from a European perspective:

- How can PbD be made a useful concept? This also leads to the next question:
- Do we let each designing party (industry and technical parts of governments) decide per casus or product line what PbD means, or is there a need for some government involvement, for example to guard some definitions and to help define PbD per application domain?

This paper describes a vision for the second approach of the second question: how the general concept of privacy by design can be translated to specific application domains in a way that is transparent and practical. In this effort, progress is the goal, not perfection. The steps that are needed are

- (1) Common understanding of the key concepts involved;
- (2) Clear, workable definition of privacy by design;
- (3) Set of tools and / or methods to give substance to privacy by design.

This vision paper follows this structure. The intended audience is quite broad: policy makers, system designers, legal and sociological scientists. This implies that we cannot go too deep into each topic, and that each type of audience may feel that too many basics have been included. This approach is however necessary to obtain common understanding through this medium.

2 Common Understanding of Key Concepts

The presence of common understanding in the key concepts is a requirement to build trust upon: privacy, trust, design and the life cycle of a system.

⁴ We have accepted the risks that come with the heavy use of IT systems. That is not the same as trusting them.

2.1 Privacy

From the point of view of a designer, a sort of checklist to verify whether his system potentially violates the privacy of data subjects is desirable. However, privacy is a broad, abstract, and subjective concept and its meaning depends on context scope, and culture. As such, it is hard to define. These are some examples in recent literature and reference works [21, 22, 23, 24, 25], one of which we repeat here: privacy is the ability to control and limit physical, social, psychological and informational access to the self or one's group [26]. However, from these definitions we learn that privacy is considered a right, a freedom, a capacity, a claim and an ability. Apparently it is a concept that is hard to capture in a single complete definition. Langheinrich gives a short history of the concept [27], and illustrates the origination of five specific categories of privacy that together appear to encapsulate all previous definitions:

- Privacy of personal behaviour (media privacy);
- Privacy of territory (territorial privacy);
- Privacy of the person (bodily privacy);
- Privacy of personal communications (interception privacy), and
- Privacy of personal data (data or information privacy).

2.2 Privacy Invading Activities

From a (US) law perspective, Solove [28] describes a taxonomy of invasions of privacy, with a collection of activities that potentially interfere with one's privacy, grouped into four categories. To do this, he illustrates the concept of privacy with a very basic system design, see Fig. 1. Solove mentions the data subject and the data holder, and describes all encountered kinds of privacy invasions in those terms.

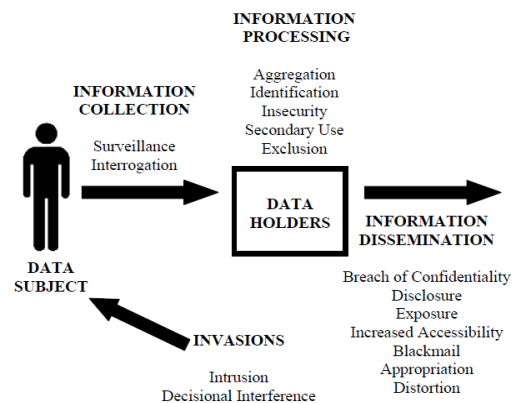


Fig. 1. - 16 potential privacy invasions (Solove, 2006)

There may be other models that are more extensive, specific or appropriate for the EU perspective. The expression of this model in information flows between entities and the completeness makes it accessible from the point of view of a designer. A model that separates all possible concerns involved helps as a checklist, which

makes it very practical and readily useful in a design process. The explicit mapping on the data subject and data holder of the potential privacy invasions can easily be used to map those invasions also into other views in system design such as behavioural (use case diagrams, sequence diagrams) and structural (composition, deployment) as can be done with UML [29].

The five categories that Langheinrich described can be mapped onto Solove's system design according to Table 1.

Table 1. -Mapping of Langheinrich's categories on Solove's privacy invasions

Solove	Langheinrich
Information collection: surveillance	Privacy of personal behaviour (media privacy); Privacy of personal communications (interception privacy)
Invasions: intrusion	Privacy of territory (territorial privacy); Privacy of the person (bodily privacy)
Information collection, Information processing and Information dissemination	Privacy of personal data

This short mental exercise suggests that a formal and complete definition for privacy may be possible.

2.3 Trust

The impact of applying privacy by design to an engineering process, should be a minimum "amount" of trust from data subjects in the new system with regard to the protection of their privacy. As Solove's model clearly illustrates, information flows one way: from the data subject to the data holder. Information is power, so there is fundamentally an imbalance of power in the relation between the data subject and the data holder which erodes trust. The process of privacy by design should be such that the interaction between all these parties leads to trust on all six points described below for the relation between data subject and data holder. This situation is complicated because in the design of systems there are more parties involved than just the data subject and the data holder: the user, designer, installer, maintainer, etc. The relation between the data subject and the data holder is the primary relation where trust should flow, but this depends on the (perceived) trust between the designer and data holder.

Trust may have even more definitions than privacy [30]. Just like in the virtual communities of Ridings et al [32], the data subjects, data holders and system designers involved in the design and use of systems generally do not converse directly with each other, so trust is at the generalized, collective level. Ridings et al asserts that trust consists of three factors: ability, benevolence and integrity. Ability is skills or competencies that enable an individual to have influence in a certain area. Benevolence is the expectation that others (i.e. trusted parties) will have a positive orientation or a desire to do good to the trustee. Integrity is the expectation that another will act in accordance with socially accepted standards of honesty or a set of principles that the trustor accepts, such as not telling a lie and providing reasonably verified information. In the context of PbD, these could be explained as described in Table 2.

Table 2. - Three factors of trust in the relations between data subject, data holder and designer

	Ability	Benevolence	Integrity
Data-subject→data holder	The data holder has the skills to protect my privacy.	The data holder is concerned about my privacy.	The data holder acts in accordance to the written and unwritten rules w.r.t. my privacy.
Data-holder→designer	The designer has the skills to design a system according to our needs.	The designer is concerned about the privacy impact of his design.	The designer acts in accordance to the design rules and best practices in this domain.

Ridings et al describe objective and measurable criteria to assess these different factors of trust in a particular community [32]. Let's focus on benevolence of the designer w.r.t. the privacy impact of his design. One criterion of trust might be “*who initiates the discussion of potential privacy risks of a particular future technology?*” Another criterion might be “*in which phase of the design are privacy concerns taken into account?*” In the public discussion around the use of UAV's for domestic surveillance, a privacy advocate might ask “*if even such an invasive tool can be made ‘privacy by design’ after it has been designed in the first place, what then, is the value of PbD?*” This is a clear sign of a lack of trust of the part of the privacy advocate in the designers of such systems. System designers and (future) data holders can address this issue by pro-actively communicating their worries about particular technologies, as head of Google Eric Schmidt recently did about drones [31].

2.4 The Law

Underlying these factors of trust, there is the legal basis, the way that a society has agreed upon to interact with each other. In Europe, the ECHR, Art.8 [32] addresses privacy. The data protection directive of 1995 is about personal data protection, which is a subset of privacy (e.g. excluding Solove's *invasions*). A directive must be integrated in the national legislative body of all EU Member States. Hence, organisations operating on the European markets face a diverse set of implementations of the European Data Protection Directive [34]. This has some disadvantages. It makes it more difficult for citizens and consumers to understand how their privacy is protected in other member states. It requires more investments by industry to tailor products and services for each member state, and multinational corporations are forced to incorporate multiple, perhaps conflicting, privacy policies within one organisation. Addressing these issues, the new Data Protection Regulation becomes immediately enforceable as law in all member states simultaneously without the necessity to be transposed into national law. This ends the numerous interpretations of the Directive in the member states.

In the domain of security, there will probably still be a Directive [35], not a Regulation. So in that domain, the EU KP7 SMART project [36] investigates some form of template structure for laws that deal with privacy-by-design. This might be helpful in keeping and building recognisability from the point of view of citizens, and therefore trust.

2.5 Design

Of the four definitions that the dictionary holds for “design”, three are relevant for the notion of PbD⁵. The first is design as noun, i.e. *a plan or blueprint*. The second is design as a verb, i.e. *creating the plan or blueprint* and refers to the early phases of the life cycle of a system. The third is also design as a verb, but with the meaning *to intend*. [37] All three definitions are relevant because an intentional design process is necessary to keep and earn trust in the resulting blueprint and its application.

When the design process becomes complex or large in some aspect, it is often called *systems engineering*. This introduces the concept of the *system*. According to the International Council on Systems Engineering (INCOSE),

"a system is a construct or collection of different elements that together produce results not obtainable by the elements alone. The elements, or parts, can include people, hardware, software, facilities, policies, and documents; that is, all things required to produce systems-level results. The results include system level qualities, properties, characteristics, functions, behaviour and performance. The value added by the system as a whole, beyond that contributed independently by the parts, is primarily created by the relationship among the parts; that is, how they are interconnected." [38]

The notion of a system also sets boundaries around it – its scope. This implies that PbD is limited to the boundaries of a system, and that these boundaries of the system should be described in order to be able to understand what the scope of PbD is in that particular instance. This hints that if a particular subsystem is designed according to the principles of PbD, whichever they may be, this subsystem can still be violating privacy laws when it is part of a broader system or connected to other systems⁶. The consequence is that a system that is designed according to the principles of PbD can still violate privacy laws when used improperly. Not only do we need to describe the limits of the system, we also need to describe and regulate its use.

2.6 Privacy and Process Models to Describe the Life Cycle of Systems

A process model for the life cycle of systems is a useful simplification of the inherently stochastic and sometimes unpredictable life cycle phases of a system. Such models range in detail, scope, track record and in flexibility, among other aspects. A good model for a design process covers all relevant phases of a system's life cycle,

⁵ The fourth definition is *a pattern used to decorate something*. This could be useful for reducing the sense of invasiveness of a design [50].

⁶ E.g., the apps on mobile phones and on social network sites can drastically alter the privacy impact of those systems.

and lets the designer and owner keep track of result, time and costs, in relation to requirements and constraints, even, or especially, in adverse circumstances, during those phases. A process model also serves as a template in the sense that it helps to methodologically follow each prescribed step. This enhances traceability, accountability and transparency in the design process. From the point of view of accountability, it is good practice to choose and apply a specific design process: you can verify whether the necessary steps have been taken, whether the transitions between the steps have been done under the right circumstances, and you can predict the next steps. The output of a phase is the start point for the next phase. So, if privacy requirements have been set at the beginning, then later in the process there should be a design where these privacy requirements have been taken into account. However, an actual design process is never flawless, so there may be mistakes in the resulting design.

There are many models for system life cycles, most of them originating in system engineering processes. Depending on the nature of a particular design challenge, a specific process-model is selected. For example, the Waterfall model is a simple and widely taught process model for designing systems. It consists of the phases Specification, Implementation, Integration, Test and Maintenance. However, the definition for PbD from the Commission itself references other phases in the life cycle of systems: the deployment, the use and the disposal phases. This disqualifies the Waterfall model for PbD because it does not acknowledge the importance of how the system is actually used, or of data disposal at the end of the life cycle of a system. Another issue with the Waterfall model is that it assumes that all stakeholders know what the problem is and on top of that, wastes no time deliberating different solution directions. The first phase of the Waterfall model is directly the requirements phase. The motivation for a particular solution direction stays implicit, and the Waterfall model is therefore not transparent. This is a risk for keeping and gaining trust. Another issue is that of repurposing a system. In a sense, scope creep is recycling, which can be beneficial from environmental and economic perspective. From the point of view of trust and transparency, we need methodological approaches to changing purposes of existing systems. The Waterfall model also lacks these.

There are other methods such as SIMILAR [39] and TOGAF [40] that do not ignore these phases of the life cycle of existing systems. The S of SIMILAR stands for State the Problem, and both SIMILAR and TOGAF have a cyclical structure, which addresses repurposing of systems. It goes too far to specify a particular design process model for PbD, but it is important to be clear about the process that is being followed, and to have a process that addresses the full life cycle of a system. A process model however, is an empty shell without the design patterns to apply it to.

2.7 Privacy and Design Patterns

With regard to PbD, the concept of design patterns is quite elegant. It does not mention implementation details, while at the same time describing the relevant aspects: problem, solution and consequences. A further, quite interesting property, is that it is for design patterns not necessary for an implementation to *exist*. This makes it

possible to also describe and register *future* technologies as design patterns. A design pattern can be considered good, or even *best practice* if it is agreed to have a particularly good track record.

Design patterns were first introduced in the domain of building architecture [41], but they became known to the information processing communities when they were introduced in IT and Object Oriented software design [42]. A design pattern is an abstraction of a design, in the sense that it is not concerned with implementation details. However, from the point of view of system design, design patterns exist at different levels of abstraction of the system. This can be illustrated with four abstractions of the same part of a house of which one function is to deliver privacy: an outer wall. A well placed wall can shield the data subject from (the feeling of) invasion, and can also actually prohibit information collection, as visualised in Fig. 2.

- A wall is a solid structure that separates outside from inside, on one floor-level; (physical view)
- A wall can be built with masonry; (design view)
- A wall can be used to shield against information collection; (use view)
- The strongest variety of this brickwork pattern to resist winds blowing straight into the wall, is *Flemish Bond* with headers every 5th row. (performance view)

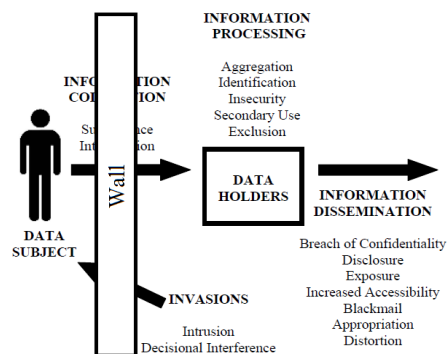


Fig. 2. A wall shields the data subject against invasions and information collection

As shown in this example, a design pattern can be described in different system aspects: the physical layout, the use case, the functional (as in information flow) design and all other system aspects or system views. Whether they are documented and intended as such is not relevant.

An example of a privacy design pattern that is about information processing is illustrated in Fig. 3. In this functional decomposition (not a process flow!) we point out that information and information processing that works at levels 0 (signal) and 1 (entity), by definition could contain personal data, while other layers should not, because they are by definition not concerned with individuals. Processing at level 2 (situation) determines “what” is happening. And finally, only processing at level 3 could make the decisions with regard to proportionality⁷, because that is where the (potential) *impact* of the situation, and therefore also of (not) doing something is taken into account.

⁷ Note that this design pattern does not say anything about whether processing is being done by an automated, or a human agent.

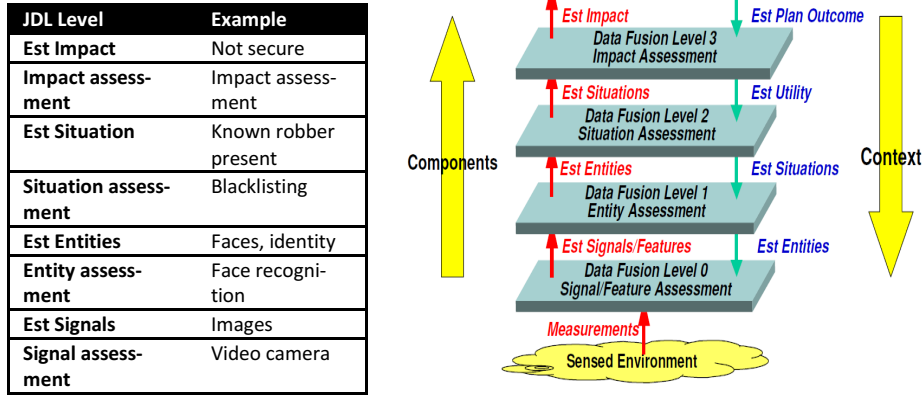


Fig. 3. -Data abstraction layers [43]- personal data resides at levels 0 (signal) and 1 (entity). The proportionality equation is done at level 3: impact assessment.

The level of abstraction of a design pattern determines whether a design pattern is relevant for privacy. This level of abstraction should be the same as the elements that are relevant for privacy: the data subject, the data holder and the privacy risks to be mitigated. In the example of the outer wall, the design view (how to build it) is not relevant from the point of view of privacy. The physical view as described is still incomplete because it still allows transparent walls, which would feel invasive and allow surveillance. So, to make the design pattern a *privacy design pattern*, it would have to be changed as follows: the “privacy-wall” is a solid opaque structure that separates the data subject from a data holder, *specifically in terms of information collection*. How this particular type of wall implements the shielding against information collection, is not mentioned. A different design pattern could be envisioned that puts the wall around the data subject *and* the data holder instead, thereby allowing data collection, but preventing *information dissemination*.

3 Extended Definition of Privacy by Design

Based on the discussion of related concepts above, we propose the following extended definition of PbD:

The principle of ‘Privacy by Design’ envisions that privacy and data protective measures are operative throughout the entire life cycle of technologies: from the early design stage to their deployment, use and ultimate disposal. This is done by applying a design process that covers all life cycle stages and by applying privacy and data protection design patterns which are well understood and are the known best-practice for the particular purpose they are used for, and domain they are used in. The resulting design documents and systems should limit all the privacy invading activities to the minimum according to the foundational principles of privacy by design.

A system designed according to this approach would score good on a PIA, and would be very easy to certify on any privacy norms. This definition links the definition as used in EU legislation with the foundational principles of PbD. It makes explicit what is expected from designers, and how traceability is organised through design methods.

3.1 Privacy Design Patterns

This definition requires agreement in a domain on what the respective best practices are for design patterns. Most design patterns that have been published so far are applicable in the domain of software engineering, and many security design patterns have also been published. Examples of a few very specific privacy design patterns can be found in [44]. A large collection –and structuring– of ICT Security design patterns is described in [45]. This paragraph introduces a more generic set of Privacy Design patterns that may be the start of a more complete set:

- privacy requirements patterns;
- anonymization and pseudonymization;
- hiding of personal data;
- data minimization;
- transparency, auditing and accounting patterns;
- informed consent.

Since for every engineering effort the system requirements are the starting point, we argue that *privacy requirements* should form an integral part of such system requirements. And since the expression of user needs in requirements in itself can be seen as a pattern, we propose the following first set of patterns that help describe the privacy requirements and select the appropriate set of controls that ensure (or make it likely) that the privacy requirements are met: *privacy needs identification*, *vulnerability assessment*, *privacy threat assessment*, *private information valuation*, *determination of privacy risks* and *selection of privacy controls*. The first five of these six patterns would combine to a Privacy Impact Assessment (PIA).

We continue with the description of a second set of privacy design patterns related to *Anonymization and Pseudonymization*. These are important and often recurring patterns in the privacy domain: they relate to the removal of the identity of the data subject in the data, or the replacement of the data subject's identity by an anonymous identifier: the pseudonym. These patterns can be recognized in many cases, e.g. the use of a pseudonym by a user of an internet forum, or the anonymization of data as is done in e.g. the voting process which is by law required to be completely anonymous. Patterns in this category are: *aggregation*, *k-anonymity*, *pseudonymous email*, *revocable privacy*, [47] *blur (part of) image* [48, 49] and *decrease time resolution*. A third set of privacy design patterns can be found around the *hiding of personal data*, such as different types of *encryption*, *batched routing* and *morphed representation*. The fourth set of design patterns we propose, revolve around *data minimization*. The idea here is to limit risks for disclosure of PII by limiting either the contents or the size of the data collection. The fifth set of design patterns has to do with methods that improve transparency and show – or even better: prove – to the data subject or to parties acting on his behalf, that due care has been taken: *logging*, *publication*, *peer review*

and *right of inspection by data subject*. The last set of privacy design patterns we discuss here, is a set of patterns around informed consent. These describe patterns that may be used to inform the data subject, and to get his or her consent for the particular purpose and method of private data processing and/or storage: *opt-in* and *opt-out*, *notification sign* and *privacy statement*.

3.2 Use Cases

In the article “Engineering Privacy by Design” [12] Gürses et al also identify the problem of vagueness of the definition of Privacy by Design, and they posit *data minimization* as a “*foundational first step to engineer systems in line with the principles of privacy by design*”. From this perspective two case studies are described that show how privacy preserving techniques may be applied to design privacy friendly systems for two applications: (1) an e-petition system, where the identity of the data subject needs to be concealed but the transaction content needs to be disclosed, and (2) an electronic toll pricing (ETP) system where the identity is disclosed but the transaction data (being data subject locations) remain concealed.

Looking at these use cases, we can easily discern which of the privacy design patterns described above, have been applied here. For both of the use cases, we recognize the following design patterns: *identification of privacy needs*, *privacy threat assessment*, *private information valuation*, *vulnerability assessment*, and *determination of privacy risks*. These design patterns emerge in the analysis the authors have done in order to derive the privacy requirements.

For the first use case, that of the privacy-friendly e-petition system, we recognize the use of *encryption* in order to provide anonymity. Furthermore, *claims* are used (“verified credentials”) instead of ordinary names or identifiers. The *publication* patterns is used to improve transparency. And finally, the authors employ *Layered Encryption* (via the use of the TOR network) in order to achieve anonymous communication.

The second use case, that of the privacy-friendly electronic toll pricing system, naturally uses a different set of design patterns because it has different privacy goals. In this use case, we see: *subsidiarity* (by choosing to not store all of the location data in one single database, but rather to leave it in the on-board units) and *cryptography*, through the cryptographic commitments.

We argue that, in addition to these design patterns, a more clear choice of the design methodology, and of the application of the other seven foundational principles of PbD would benefit the privacy of data subjects in these use cases. For example, the end-of-life phase is also ignored by Gürses et al.

3.3 Supporting Tool and Methods

In the areas of software engineering and of security, rich collections of design patterns already exist. Large online communities contribute to the descriptions of existing and new design patterns. Such communities also arise for privacy design patterns [46]. Published and reviewed collections of design patterns may help a system designer

determine which privacy design patterns (PDP) are best practice, and it may help data subjects to recognize the signs of good implementations of such design patterns. A good knowledge base contains a body of reference of PDP's and facilitates searching from different entries:

- privacy invading activities and their associated risks which are to be mitigated by a PDP, along the categories of Solove and Langheinrich, or those introduced by Van Lieshout [5], or by Cvrček and Matyás [51];
- real world instances (implementations) of the design pattern. This information would typically come from a Privacy Impact Assessment. Citizens and designers can inspect those implementations;
- consequences of applying the PDP. Citizens can recognize those consequences;
- types of systems that would benefit from applying PDP's, e.g. road pricing system, surveillance system, hospital information management system;
- manual's, documentation, publications and intellectual property that helps building the PDP's, while staying technology-, and therefore vendor-neutral;
- maturity of PDP's. It may have an indication of the *Technology Readiness Level*. Designers and procurers can choose the amount of risk they want to take with regard to new PDP's.

The tool could be implemented in the form of a website with a database behind it. This tool and methods will increase trust because it directly influences the six factors influencing trust that were introduced earlier:

- the *benevolence* of the data holder and the system designer is illustrated by how well they maintain their respective systems in the knowledge base;
- the *ability* of the data holder and the system designer is illustrated by their choice for a suitable design process and design choices w.r.t. PDP's;
- the *integrity* of the data holder and the system designer is illustrated by their participation in the knowledge base, and by following the best practices and design methodologies that are recommended for their respective domains.

4 Conclusions

Legislation makes extensive references to “privacy by design”, but does not specify what it exactly means. This omission of a clear definition of PbD in the European landscape entails that from the point of view of European citizens, policy makers, local authorities and industry there is currently an unclear situation with regard to what the implications of a request for PbD actually entails. For example, it should be investigated whether the value of PbD can be improved by including the roles of designer or builder of a system in Data Protection Laws.

It would be elegant if individual and collective privacy interests could be aligned with economic interests. However, citizens currently do not flock to products and services that better protect their privacy, so there is no market mechanism favouring privacy protecting design. If we cannot leave it to economics to protect data subject's privacy, then we may need a more regulated approach. Already, the proposal for a

(new) privacy regulation creates a clear economic motivation, at least for enterprises, to address privacy risks. Privacy by Design could be the label of the method that describes how to mitigate such risks. At the same time, technology advances, the context changes and our knowledge improves. So, in addition to regulation, designing *privacy by design* should be an on-going, transparent dialogue between representatives of data subjects, data holders and system designers.

In order to facilitate this dialogue, some terminology must be defined more clear. The five categories of privacy that we encountered in literature can be mapped on Solove's system design which suggests that Solove's taxonomy is inclusive enough. This helps build trust in any methodologies and tools that are based on such a taxonomy.

Applying a design process model enhances traceability, accountability and transparency in the design process. An essential element of PbD should be, that the *privacy requirements* must be clearly and early stated as part of the functional requirements. Apart from serving as guide in the implementation /construction process, they also give guidance when testing the resulting system for compliance. The design process model should further cover the entire life cycle from problem statement to system disposal, including also the use-phase: a system that is designed in line with the principles of PbD can still violate privacy laws when used improperly.

In addition to common understanding of key concepts and the application of the right process model, we propose a more coherent approach to privacy design patterns (PDP). PDP's describe technology on an abstract level, and as such can also describe both existing, and not-yet-existing technology that enhances privacy of the data subject. We suggest that, when describing a privacy design pattern, the privacy invading activities (of Solove) that have to be mitigated are also mentioned, in other words: which privacy problem is solved. The community of PbD-designers could be supported with a knowledge base of such patterns.

There may be more foundational principles than the seven of Cavoukian, e.g. data minimisation could be another. An extended definition of PbD has been given that links the definition as used in EU legislation with the foundational principles of PbD. It makes explicit what is, in a certain domain, expected from designers, and how traceability is organised through design methods. This approach can be applied to all domains, as long as the PDP's are considered best practice in the respective domains. This implies that intrinsically intrusive domains such as surveillance can also benefit from a PbD-approach.

A tool and a set of use cases for that tool are envisioned to approach PbD with the help of privacy design patterns and design process models. Different types of stakeholders can address their interests with the tool in different design phases. It can be used by designers, citizens, DPAs and policy- and decision makers, among others. This tool would expand the level of transparency from *what data is being collected*, to *how are we protecting your privacy?*

The new privacy regulation introduces sanctions up to 2% of the annual turnover of enterprises. This elevates the importance of mitigation of privacy risks. This vision paper positions Privacy by Design as the mechanism to mitigate these privacy risks, and gives practical guidelines to *design Privacy by Design*.

Acknowledgements. The authors wish to thank Dr. John Borking, Dr. Jaap-Henk Hoepman, Sander van Oort and Johanneke Siljee for their thorough reviews.

References

1. EC, Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data (1995)
2. EC, COM(2012) 11 (final) Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on the protection of individuals with regard to the processing of personal data and on the free movement of such data (General Data Protection Regulation) (January 25, 2012)
3. EC, COM(2010) 609 (final), A comprehensive approach on personal data protection in the European Union (November 4, 2010)
4. EC, COM(2010) 245 (final)/2, A Digital Agenda for Europe (August 26, 2010)
5. van Lieshout, M., Kool, L., van Schoonhoven, B., de Jonge, M.: Privacy by Design: an alternative to existing practice in safeguarding privacy. *Info.* 13(6), 55–68 (2011)
6. European Forum for Urban Security, Charter for a Democratic Use of Video Surveillance (2011)
7. Hes, R., Borking, J.: Privacy Enhancing Technologies: the path to anonymity (Revised Edition) Registratiekamer, Achtergrondstudies en Verkenningen 11 (first edition 1995)
8. CFP2000, Conference on Computers, Freedom & Privacy (2000), <http://www.cfp2000.org/>
9. EC / TNO et al, FP5, PISA project (2003), http://cordis.europa.eu/projects/rcn/53640_en.html (accessed June 2, 2012)
10. Cavoukian, Origins of Privacy by Design, <http://privacybydesign.ca/publications/pbd-origin-and-evolution/> (accessed August 3, 2011)
11. Cavoukian, Privacy by Design – The 7 foundational principles (August 2009) (revised January 2011)
12. Gürses, Troncoso, Diaz: Engineering Privacy by Design. In: Conference on Computers, Privacy & Data protection, CPDP (2011)
13. Jean-Philippe Courtois, Privacy by Design at Microsoft (November 29, 2010)
14. Winterfield, K. (2009), <http://ibmresearchnews.blogspot.com/2009/10/inventors-corner-innovations-enable.html>
15. Cavoukian, Privacy by Design – The answer to overcoming negative externalities arising from poor management of personal data, Trust Economics Workshop (June 23, 2009)
16. Kranzberg, M.: Technology and History: Kranzberg's Laws. *Technology and Culture* 27(3), 544–560 (1986)
17. EuroPrise - the European Privacy Seal for IT Products and IT-Based Services (2007), <https://www.european-privacy-seal.eu/> (accessed June 2, 2012)
18. London Economics, Study on the economic benefits of privacy-enhancing technologies (PETs) (July 2010)
19. Borking, J.: Privacy law is code (2010)
20. Rogers, E.M.: Diffusion of Innovations (1962)

21. Warren and Brandeis, Harvard Law Review. The right to privacy, vol. IV(5) (December 15, 1890),
http://groups.csail.mit.edu/mac/classes/6.805/articles/privacy/Privacy_brand_warr2.html
22. Agre & Rottenberg, Technology and privacy: the new landscape (1997)
23. Clarke, R.: Roger Clarke's 'What's Privacy?',
<http://www.rogerclarke.com/DV/Privacy.html> (accessed May 12, 2011)
24. Cambridge Essential English Dictionary, lemma Privacy (accessed August 6, 2011)
25. Westin, A.: Privacy and Freedom. Atheneum, New York (1967)
26. Burgoon, K., Parrott, R., Le Poire, B.A., Kelley, D.L., Walther, J.B., Perry, D.: Maintaining and Restoring Privacy through Communication in Different Types of Relationships. *Journal of Social and Personal Relationships* 6(2), 131–158 (1989)
27. Langheinrich, M.: Privacy by design - principles of privacy-aware ubiquitous systems. In: Abowd, G.D., Brumitt, B., Shafer, S. (eds.) *UbiComp 2001*. LNCS, vol. 2201, pp. 273–291. Springer, Heidelberg (2001)
28. Solove, D.J.: A Taxonomy of Privacy. *University of Pennsylvania Law Review* 154(3), 477–564 (2006)
29. UML 2.4.1 Specification, <http://www.omg.org/spec/UML/2.4.1/> (accessed December 2011)
30. Harrison McKnight, D., Chervany, N.L.: The Meanings of Trust, University of Minnesota (1996), <http://www.misrc.umn.edu/wpaper/wp96-04.htm>
31. BBC, Eric Schmidt, Google (April 13, 2013),
<http://www.bbc.co.uk/news/technology-22134898>
32. Ridings, C.M., Gefen, D., Arinze, B.: Some antecedents and effects of trust in virtual communities. *The Journal of Strategic Information Systems* 11(3-4), 271–295 (2002) ISSN 0963-8687, 10.1016/S0963-8687(02)00021-5
33. Article 8 of the European Convention on Human Rights (1950)
34. EC, undated, Status of implementation of Directive 95/46 on the Protection of Individuals with regard to the Processing of Personal Data (2011),
http://ec.europa.eu/justice/policies/privacy/law/implementation_en.htm (accessed August 3, 2011)
35. EC, COM/2012/010 final - 2012/0010 (COD), Proposal for a Directive of the European Parliament and of the Council on the protection of individuals with regard to the processing of personal data by competent authorities for the purposes of prevention, investigation, detection or prosecution of criminal offences or the execution of criminal penalties, and the free movement of such data (January 2012)
36. EU KP7 SMART project, <http://www.smartsurveillance.eu/> (accessed May 13, 2012)
37. Cambridge Essential English Dictionary, lemma Design (accessed August 28, 2011)
38. INCOSE, A Consensus of the INCOSE Fellows,
<http://www.incose.org/practice/fellowsconsensus.aspx>
 (accessed June 2012)
39. Bahill, A.T., Gissing, B.: Re-evaluating systems engineering concepts using systems thinking. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 28(4), 516–527 (1998)
40. The Open Group, "The Open Group Architecture Framework, TOGAF",
<http://www.opengroup.org/togaf/> (last accessed April 2, 2012)
41. Alexander, C.: A Pattern Language: Towns, Buildings, Construction (1977)

42. Buschmann, F., Meunier, R., Rohnert, H., Sommerlad, P.: Pattern-Oriented Software Architecture. A System of Patterns, vol. 1. John Wiley & Sons (1996)
43. Steinberg, A., Bowman, C.: Rethinking the JDL Data Fusion Levels, NSSDF JHAPL, June, 04 2. In: Bowman, C.L. (ed.) The Dual Node Network (DNN) Data Fusion & Resource Management (DF&RM) Architecture, AIAA Intelligent Systems Conference, Chicago, September 20-22 (2004)
44. Hafiz, M.: A collection of Privacy Design Patterns. In: Proceedings of the 13th Pattern Languages of Programs. Allerton, Illinois (2006)
45. Security Patterns – Integrating Security and Systems Engineering, Schumacher, Fernandez-Buglioni, Hybertson, Buschmann, Sommerlead. John Wiley & Sons (2006)
46. UC Berkeley School of Information (2013), <http://privacypatterns.org/> (last visited May 2013)
47. Revocable Privacy, Jaap-Henk Hoepman. Privacy & Informatie 11(3), 114–118 (June 2008)
48. BSIA, Privacy Masking Guide (2011)
49. Roelofsen, Patent WO 03/010728/A1 Method and System and Data Source for Processing of Image Data (February 2003)
50. WeArePerspective (2007), <http://www.weareperspective.com/project/ns-camera> (accessed December 2011)
51. Cvrček, D., Matyáš, V.: D13.1: Identity and impact of privacy enhancing technology. FIDIS (2007), <http://fidis-wp13-del13.1.final.pdf> (accessed February 16, 2011)