The Current State of the Holistic Privacy and Security Modelling Approach in Business Process and Software Architecture Modelling

Sascha Alpers[1], Roman Pilipchuk[1], Andreas Oberweis[1] and Ralf Reussner[1]

[1] FZI Forschungszentrum Informatik, Haid- und Neustraße 10-14, 76131 Karlsruhe, Germany
{alpers, pilipchuk, oberweis, reussner}@fzi.de

Abstract. Modelling is central for business process and software architecture documentation and analysis. However, business processes and software architectures are specified with their own highly developed languages, methods and tools. There are approaches in the literature for modelling privacy and security issues using existing business process or architecture modelling languages to express different requirements by enriching these languages with annotations. Nevertheless, there is a lack of formalization and therefore the potential use for tool-based analyses are limited. In addition, the continuity between business and software models is not granted, but when modelling compliance requirements like privacy, traceability is very important, e.g. for compliance checks. In this contribution, approaches for modelling security and privacy in business and software models are examined. One key finding is that there is currently no comprehensive modelling approach which covers the necessary aspects and perspectives. This could include processes as well as, for example, organizational and data structure questions. In conclusion, we suggest developing a new holistic modelling approach which includes the needed aspects and with a concept for the traceability of the requirements from business models to software architecture models.

Keywords: Business Architecture, Software Architecture, Modelling, Privacy.

1 Introduction

Many companies, especially large companies, model their organizational processes and software systems. This is to define and improve them, identify and reduce flaws. Explicit models of processes and software architectures not only enable their analysis and optimisation, these models also save costs during the evolution of processes and software architectures. However, business and software system experts typically use different modelling languages. There exist many languages for modelling business processes. BPMN, a semi-formal notation, is the most prominent one. Petri nets provide a formalized view of processes. Transformations which establish mappings between BPMN and Petri nets exist. In the following, we focus primarily on Petri net (Reisig,

2013) models and consider BPMN only marginally. The state-of-the-art modelling language for software systems is UML (OMG, 2017). As neither business process modelling languages nor UML have elements capable for modelling privacy, extension mechanisms exist for introducing additional symbols to model various aspects of privacy. Additionally, security is also relevant because privacy is related to some security goals such as confidentiality or integrity. Both security and privacy are becoming increasingly important, for example due to the upcoming General Data Protection Regulation (GDPR) (European Union, 2017).

Although there are many approaches to extending business process modelling notations and UML to cover security and other aspects, there is no common and generally accepted approach for modelling privacy. A broad variety of approaches exists for introducing additional symbols to model privacy directly or indirectly, through security elements. However, the extent to which privacy can be modelled depends on the proposal. Additionally, modelling approaches which support transformations from business process models to software design to keep business process models like Petri nets and software models like UML consistent with each other are missing. Due to these reasons, we analysed the capabilities of existing software architecture-oriented and business process-oriented modelling approaches to model privacy aspects. We analysed, how privacy can be modelled and investigated the possibility of and need for a comprehensive modelling language in the field of privacy to cover business processes and software systems. We selected these approaches according to their abilities to model privacy aspects directly or indirectly, through security aspects. The selected approaches were analysed and compared with each other to identify their similarities and differences. This was done to understand the need for a comprehensive model of privacy aspects and to explore how it could be realized, beginning from a business process model and then leading to a software architecture model. For this, we categorized the approaches and identified two criteria, namely "security mechanisms" and "different views". "Security mechanisms" describes the elements and mechanisms by which the approach supports privacy modelling. The second criterion, "different views", groups approaches according to the view of the stakeholder for whom the approach is intended. Our results show that only a few approaches actually introduce elements to model privacy principles. In the following Section 2, we describe why the needs for a holistic modelling approach is increasing. Section 3 presents the business process-based approaches. Software architecture-based approaches are presented in Section 4. Section 5 discusses similarities and differences between both approaches. The contribution ends with some concluding remarks in Section 6.

2 Increasing Need for Holistic Modelling

In the past few years, companies have faced the increasing problem of cybercrime (Accenture, 2017). Cybercriminals are becoming more organized and cooperating in larger groups, allowing them to undertake more and more complex attacks. Companies also face a growing number of security laws with which governments require them to comply. Especially companies that operate globally have to comply with the laws of different countries. To state some of them, the Basel Accords and Minimum Requirements for Risk Management (MaRisk) (Federal Financial Supervisory Authority, 2005) regulate the risk management for the finance sector; the IT Security Act (Federal Office for Information Security, 2015) regulates the security of IT systems for critical infrastructures; and the General Data Protection Regulation (GDPR) (European Union, 2017) governs data collection, processing and the use of personal data in the European Union. However, privacy regulation is not new. In 1970 the first formal worldwide data protection law came into force in the German federal state of Hesse (Genz, 2004), in 1984 the *Bundesverfassungsgericht* (Federal Constitutional Court) created the basic right of informational self-determination based on the general right of personality (Art. 1(1) and Art. 2(1) German *Grundgesetz* [Basic Law]) (Hornung and Schnabel, 2009) and in the European Union, a 1995 European directive set the framework conditions for the processing of personal data. But the GDPR imposes financial penalties of up to four percent of an organization’s worldwide turnover, which is similar to other regulations.

The business of companies is becoming more complex every year. Supply chains and manufacturing are increasingly distributed all other the world and operate in complex ecosystems. Thus, companies face the complicated task of developing rules and standards in order to protect their sensitive personal datadata and business secrets according to their needs. They are of the utmost importance, as only the business level of a company knows which data are critical and their required level of protection. Altogether, we see that IT security is becoming more and more crucial for companies of all kinds. That is why the business level is charged with several additional goals pertaining to IT security. Firstly, to prevent cybercriminal attacks, reputational damage and consequently the loss of monetary income, they have to establish organization-wide IT security. There are various guidelines like the ISO/IEC 27000-series (International Organization for Standardization and International Electrotechnical Commission, 2014) or the IT Baseline Protection (German Federal Office for Information Security, 2006) which describe how to establish, manage and maintain information security effectively in organizations. Access control requirements from the business level perspective are described there too. Guidelines like ITIL (AXELOS, 2011) or COBIT (Information Systems Audit and Control Association, 2012), which comprise sets of practices for IT service management, introduce dedicated business processes for IT security and access control. Therefore, establishing organization-wide IT security is a complicated task involving different departments and various models. Secondly, during the establishment of organization-wide IT security, companies have to comply with an increasing number of security laws. This means that the compliance department is a fundamental part in the whole process. Thirdly, as only the business level knows which assets need to be protected, they have to define the rules and standards on how to interact with these assets. To sum up, the business level in a company becomes a key point in establishing security and privacy and therefore has to work closely with many different departments like IT and compliance departments, resulting in diverse models relevant for IT security and privacy. Thus, there is a need for a systematic transformation between these models to keep them consistent with one another. Only in this way can a good alignment can be realized.

IT security and privacy has become crucial for all kind of companies. One thing IT security and privacy have in common is the need for access control requirements. Both IT security and privacy impose access restrictions on certain data. While IT security describes principles, algorithms and protocols on how to restrict access, privacy describes who should have access to which personal data and how to handle it. These access control requirements come partially from security laws and security guidelines. The business level establishes the other part in terms of rules and standards, as described above. They are both modelled increasingly in business processes, due to the obligation or decision of companies to implement IT service management guidelines like ITIL or COBIT. IT departments must adapt these access control requirements such as enterprise architectures, system architectures and so on in their own models. A typical modelling language here is UML [(OMG, 2017), (Störrle, 2017)]. Different knowledge about terminology is a problem and creates a communication gap that opens up the potential for errors. This poses a severe problem, because any error can undermine security. Thus, both the IT department and the business level have an interest in keeping their numerous models consistent, so that access control requirements are implemented correctly and consistently.

Often, the fact that companies are evolving is neglected. This means that systems, requirements, business processes, enterprise architecture and other models steadily evolve. They all have a lifecycle and affect each other in non-trivial ways (Aerts, Goossenaerts, Hammer and Wortmann, 2004). Their complex interrelations are not understood well and have not yet been adequately researched (Aerts, Goossenaerts, Hammer and Wortmann, 2004). As stated above, problems here may lead to security breaches. Hence, there is the need for a fast and automatic transformation between the models to keep IT security and privacy information correct and consistent. Additionally, it is important to understand the mutual dependencies so that the various departments can react to changes. Traceability between the models can help, since it allows tracing and understanding design decisions. Both traceability between business and IT models and their mutual interdependence are not yet well researched.

Access control requirements formulated in law and in guidelines must be incorporated and extended by the business level and then implemented by the IT department. There is a need for a transformation between all models of the involved parties. Considering the increasing number of companies implementing guidelines like ITIL and COBIT, as well as the close collaboration between the business level and the compliance department, business processes today comprise many access control requirements. These business level access control requirements represent the demands of law. A promising way to close the gaps described above would be to extract the access control requirements from business processes and transform them to the various models of the IT. Enterprise architectures offer the right granularity and could be analysed as to whether they comply with the extracted access control requirements by using a data flow analysis. Another possibility is to transform the access control requirements directly into permissions for an access control system. Clearly, the increasing need opens a large and promising field of research for transformation and consistency problems between models of different areas.

3 Software Architecture-oriented Approaches

This chapter introduces the software architecture-oriented approaches for modelling privacy. The first section gives a brief introduction to the de facto standard modelling language in the field of software engineering and the second section is an inspection of the architecture-based approaches in the context of privacy and confidentiality.

3.1 Modelling

The Unified Modelling Language (UML) is the current standard for modelling architecture in software engineering. De facto UML is a general-purpose language which is standardized by the Object Management Group (OMG). It comprises 14 diagrams divided in two major diagram types: structure diagrams and behaviour diagrams (OMG, 2017). While structure diagrams mainly focus on illustrating the static structure of a system, behaviour diagrams point out its dynamic part. The sequence diagram shows the chronological flow of messages between objects. It brings an additional technical dimension to the practice and is an integral part of the described static structure. The use case diagram visualizes functional requirements, including the different actor groups and their suitable participatory methods or relationships. Class diagrams describe classes, associations, methods and their attributes. This is a short overview of the modelling diagrams in UML. A detailed explanation can be found in the UML specification (OMG, 2017).

3.2 Analysis of Software Architecture-oriented Approaches

This section surveys the software architecture-based approaches. Table 1 summarizes all analysed papers, the types of UML diagrams used, whether they extend through UML profile or not, and what the extension allows to be modelled.

(Jutla, Bodorik and Ali, 2013) propose an extension to the UML use case diagram for representing privacy specifications like pseudonymization, anonymization and consent in an easily understandable way (see Table 1 no. 1). The extension is not based on the UML profile extension mechanism. Instead, a Microsoft Visio extension ribbon is created that offers the required elements. All possible privacy requirements and specifications can be expressed due to the use of free text fields. Furthermore, in use case diagrams the extension works by introducing a ‘super container’ in-between actors and use cases. Privacy control classes and obligations are stated inside the super container. This extension enables it to express all kinds of privacy principles and allows a technical specification of other security principles like confidentiality. (Basso, Montecchi, Moraes, Jino and Bondavalli, 2015) introduced a UML profile which is capable of expressing different privacy concepts through privacy policies incorporated in various

UML diagrams (see Table 1 no. 2). Privacy policies are composed of one or more statements which describe the rules specified in the privacy policy. Besides that, they also specify the purpose of data collection, its management, and the prerequisites that need to be met. Private data and actions performed on it can be aggregated and translated into standardized stereotypes to, for example, identify to whom the access to private data is granted, the period, and the usage behaviour of the target groups. Several other stereotypes describe how the data are provided and managed, either by a user or by a system. In both cases, the UML profile allows the design of privacy-aware applications by modelling the application's privacy policy and keeping track of the elements responsible for enforcing it. The profile not only allows modelling of access control on private data, but also of privacy principles like consent, data security and purpose limitation.

No.	Paper	Diag. Type	Ext. Through	To Model
1	Engineering Privacy for Big Data Apps with the Unified Modelling Language	Use Case	Super container	Privacy specifications
2	Towards a UML Profile for Privacy-Aware Applications	Various	UML profile	Privacy policies
3	UMLsec: Extending UML for Secure Systems Development (+2)	Various	UML profile	Security requirements / primitives / management and threat scenarios
4	Supporting Confidentiality in UML: A Profile for the Decentralized Label Model	Class	UML profile	Decentralized label model
5	Towards the Engineering of Security of Information Systems (ESIS): UML and the IS Confidentiality	Sequence	UML profile	Access control and information flow control
6	A UML Profile for Requirements Analysis of Dependable Software	Class	UML profile	Problem frames (e.g., confidentiality, integrity)
7	Extending UML for Designing Secure Data Warehouses (+2)	Class	UML profile	Security classes and separation of duty
8	Weaving Security Aspects into UML 2.0 Design Models	Class and Sequence	UML profile	Security requirements and aspect-oriented solutions
9	CMP: A UML Context Modelling Profile for Mobile Distributed Systems	Class	UML profile	Privacy restrictions

Table 1. Overview of software architecture-oriented approaches (Alpers, Pilipchuk, Oberweis and Reussner, 2018).

(Jürjens, 2002) proposed a UML profile, called UMLSec, which is shown in Table 1 no 3. It is specifically constructed to express security-relevant information within various UML diagrams. In particular, it enables non-experts in the area of security to express their security needs easily. UMLSec enables software engineers to express basic security requirements including security concepts, security primitives, security management and threat scenarios. This allows modelling of confidentiality of information and information flows. Furthermore, it is possible to check whether the constraints associated with the stereotypes are fulfilled by a given specification and, by this, indicate possible vulnerabilities (Jürjens, 2005).

(R. Heldal, Schlager and Bende, 2004) present a UML profile with a decentralized label model incorporated into UML class diagrams (see Table 1 no. 4). This allows the modelling of confidentiality at design time. The so-called UMLs profile allows the specification of confidential information flow in a fine-grained manner. Different stereotypes defining owners and users are used to annotate classes, attributes, operations, parameters, errors, and return types. These labels are used to decide whether the information flow is permitted or not. Declassification of information is realized with the authorityConstraint. It models the weakening of the confidentiality of information coming from more confidential sources. This is necessary for operations processing confidential data but providing less confidential results. The approach is presented for class diagrams, but it is extendable to other diagram types such as interaction, use case and activity diagrams.

The work of Goudalo et al. (Goudalo and Seret, 2008) elaborates on modelling security aspects of information systems (see Table 1 no. 5). They propose a UML profile on how to properly encapsulate security knowledge during design time. An example is shown in the context of confidentiality. Confidentiality of information and information flow is modelled in sequence diagrams by defining stereotypes modelling the confidentiality levels of resources, subjects, and subsystems. In essence, software engineers are able to model confidentiality in diverse ways by using this UML profile.

Table 1 no. 6 shows the work of Hatebur et al. (Hatebur and Heizel, 2010). They build upon a UML profile for expressing problem frames in UML class diagrams. Problem frames are patterns are used to define problem classes by their contexts and characteristics. The extended UML profile expresses dependability requirements. In the case of security, the traditional goals of confidentiality, availability and integrity can be expressed. These goals are modelled with stereotypes and include specifications like the data to be secured, the attacker and the stakeholder of data. Additionally, problem frames allow the expression of arbitrary confidentiality requirements. The authors mention that the main advantage of their approach is the ability to express dependability requirements without the anticipation of a solution. This clearly separates the problem space from the solution space. Furthermore, it is easy to visually distinguish between different security requirement classes.

The approach of (Fernandez-Medina, Trujillo, Villaroe and Piattini, 2004), SECDW allows the modelling of confidentiality aspects in UML class diagrams (see Table 1 no. 7). SECDW is an extension intended for the domain of data warehouses. The approach introduces a UML profile that enables the specification of security classes for information and users. Tuples composed of security classifications, sets of user compartments (classification of users in department like structures), and user roles allows the specification of constraints about which users are allowed to read certain information. Triki et al. (Triki, Ben-Abdallah, Feki and Harbi, 2010) proposes an extension (SECDQ+) with the ability to model leaks of confidential information. Examples are health information or company turnover which, if accessed in combinations of datasets, leak additional undesired information. This problem is known as conflict of interest (Triki, Ben-Abdallah, Feki and Harbi, 2010).

The UML profile of (Mouheb, Talhi, Lima, Debbabo, Lang and Pourzandi, 2009) is capable of both capturing security requirements and specifying security solutions (see Table 1 no. 8). This is achieved by placing security aspects into UML class and sequence diagrams in an aspect-oriented modelling manner. Besides that, the approach allows the expression of the separation of security concerns for software functionalities. Security experts can specify security solutions as aspects in the UML model and model their points (where the security solutions are implemented) in UML sequence diagrams. In consequence, the solution is easily understandable even for non-security experts.

The UML profile of (Simons, 2007) models privacy restrictions in UML class diagrams (see Table 1 no. 9). The target field is in the context of mobile distributed systems, but the approach can be used in other contexts as well. The main idea is to bind access rights to context information. This is done by formulating privacy restrictions on context information. Privacy restrictions are composed of the source and validity of the context information, as well as the access rights in the form of confidentiality levels. In Simons' UML profile, constraints are used to validate the model. This is accomplished by imposing restrictions on the defined stereotypes to enforce the correct use of the profile.

4 Business Process-oriented Approaches

Privacy and security are business requirements, and therefore privacy as well as security requirements are increasingly included in enterprise modelling (Shariati, Bahmani, Shamst, 2011). This can be achieved in different ways:

- via models of privacy and security aspects using normal enterprise modelling languages
- in the form of annotations
- with the help of more-or-less formalized privacy/security notation add-ons for existing modelling languages

For business processes as one component of enterprise modelling, we analysed 'Petri nets' and 'Business Process Model and Notation (BPMN)'.

4.1 Analysis of Petri Net-based Approaches

There are plenty of approaches to using Petri nets for modelling information security aspects, particularly information confidentiality. They can be used to model privacy requirements as well, but special privacy model extensions are not common today. The problem is that some of the approaches only focus on the technical level, which generally means that they are discussing problems like algorithms, protocols or technical architecture, using Petri nets for visualisation, but omit the business process perspective.

Huang and Kirchner have introduced a formal method to verify whether the compositions of sub-policies fulfil the required general policies of a company (Huang and Kirchner, 2013). They used coloured Petri nets and Petri net-based properties like completeness, termination, consistency and confluence. One use case is the verification as to whether a set of policies fulfils a general policy like GDPR. Therefore, the requirements of the GDPR must be transformed into a model.

(Mixia, Qiuyu, Dongmei and Hong, 2005) extended object Petri nets by using modules to define security services like the decryption and encryption of data. This could be interesting for data protection because encrypted data need not be protected itself as long as the key is strong and kept secret. (Akbarzadeh and Azgomi, 2010) defined a framework for the assessment of security protocols. They used coloured stochastic activity nets and implemented probabilistic model checking. In addition, (Bouroulet, Devillers, Klaudel, Pelz and Pommereau, 2008) analysed security protocols and a Petri net extension called S-net, which is designed such that the terms of the Security Protocol Language (Crazzolara and Winskel, 2001) can be used. Other Petri net-based approaches aim at building models for special concepts. For example, (Zhang, Hong and Liao, 2006) modelled the Chinese wall policy with coloured Petri nets; afterwards, they used a coverability graph to analyse the guarantees of the Chinese wall policy. (Henry, Layer and Zaret, 2010) used coupled Petri nets for the risk analysis of computer networks. Sun et al. published a ‘Verification Mechanism for Secured Message Processing in Business Collaboration’ (Sun, Yang, Wang and Zhang, 2009). They used the role-based access control (RBAC) mechanism and hierarchical coloured Petri nets to detect conflicts in message access within collaboration process instances to the role-based policy. A similar approach from (Lai, Hong and Jeng, 2008) focused on the confidentiality of information exchanges between organizations and therefore has special places in coloured activity nets for incoming and outgoing information. Chinese wall and interorganizational information exchange are also relevant for privacy protection questions. As shown, many approaches use Petri nets for modelling security aspects, but focus on a technical level or only cover one single aspect. Therefore, these approaches are not suitable for use by business process experts to model their security requirements and discuss them with technical experts.

In addition, some approaches use Petri nets for modelling or analysing security aspects of business processes. Accorsi and Wonnemann developed InDico (Accorsi and Wonnemann, 2011), an information-flow analysis method for labelling Petri net-based business process models. InDico focuses on ‘information propagation throughout the systems (end-to-end) rather than mere data access (point to point)’ (Accorsi and Wonnemann, 2011). Accorsi et al. (Accorsi, Lehmann and Lohmann, 2015) published an extension of InDico for analysing information-flow effects during process execution. They used security levels (called ‘levels of confidentiality’) but reduced them to two, and analysed the structural interferences between them. It is impossible to express different levels of confidentiality for the same place in one business process scheme, e.g., different information, or more than two levels of confidentiality for the whole business process scheme. Li et al. (Li, Wu and Huang, 2009) described a coloured Petri net extension for detecting confidentiality problems in information-flow models. They use security levels and add the concrete security levels as attributes of the tokens. Li et al. did not focus on the resources handling the information. Knorr (Knorr, 2001), who also used security levels, presented a method to verify multilevel security policies in workflow models, but he modelled control and information flow as different arcs in his workflow Petri nets. Atluri and Huang (Atluri and Huang, 1996), who have also used Petri nets, presented a multilevel security approach with security levels for places and tokens. They later extended their approach with more concepts, like separation of duty and role-based access, using a coloured, timed Petri net (Atluri and Huang, 2000). They did not consider resources or the possibility of reducing the security level of a token, e.g., when information is truncated.

The large number of approaches for modelling security aspects using (high-level) Petri nets shows that the integration and processing of confidential information in Petri net-based business process models is currently a major challenge. This is one reason why we think Petri nets are also suitable for privacy questions. Other reasons in favour of Petri nets are their mathematical foundation and the availability of a broad range of analysis methods. Especially for analysis functionality, formal Petri nets are necessary.

4.2 Analysis of BPMN-based Approaches

Extensions of the Business Process Model and Notation for modelling security requirements exist for each of the three classic security objectives: confidentiality, integrity and availability. Leitner et al. (Leitner, Miller and S. Rinderle-Ma, 2013) have published a systematic literature review on 'Security Aspects in the Business Process Model and Notation'. Therefore, we do not provide a detailed overview here. In summary, some publications use BPMN for security questions without new extensions. In (Meland and Gjaere, 2012), Meland and Gjaere argue that there is no need for new BPMN extensions for many questions. Several other approaches extend the BPMN notation, e.g., with new symbols to create a faster overview of security issues for the model users (Wolter and Meinel, 2010). Focusing on privacy as part of security, (Mülle, Stackelberg and Böhm, 2011) used BPMN to introduce privacy in business process models, while Labda et al. (Labda, Mehandjiev and Sampaio, 2013) extended BPMN to privacy-aware BPMN. They focused not only on modelling privacy aspects, but also proposed a methodology for transferring them into the implementation.

5 Comparing Approaches

We have identified two criteria through which the software architecture-oriented and business process-oriented approaches can be conceptionally compared. In summary, only a few approaches we reviewed introduced elements to model actual privacy principles [(Julta, Bodorik and Ali, 2013), (Basso, Montecchi, Moraes, Jino and Bondavalli, 2015), (Atluri and Huang, 2000)]. Most of them introduce privacy as a way of establishing confidentiality and restricting access to information.

5.1 Security Mechanisms

This criterion describes the expression of privacy in models in terms of how it is expressed, and through which security and privacy mechanisms it is represented. We recommend the following two characteristics for an analysis:

- Information flow and access control: this characteristic establishes privacy by introducing concepts that restrict the information flow or the access to information, functions or system parts by imposing rights. Approaches with this characteristic introduce concepts of confidentiality in various ways as well as in different degrees. These concepts are used either directly or can be used to express privacy in a certain way. Examples are Chinese wall policy and confidentiality levels. The following approaches fulfil this characteristic [(Jürjens, 2002), (Heldal, Schlager and Bende, 2004), (Goudalo and Seret, 2008), (Simsons, 2007), (Fernandez-Medina, Trujillo, Villaroel and Piattini, 2004), (Zhang, Hong and Liao, 2006), (Sun, Yang, Wang and Zhang, 2009), (Lai, Hong and Jeng, 2008), (Accorsi and Wonnemann, 2011), (Accorsi, Lehmann and Lohmann. 2015), (Li, Wu and Huang, 2009), (Knorr, 2001), (Atluri and Huang, 2000), (Mülle, Stackelberg and Böhm, 2011)].
- General structures: approaches with these characteristics use abstract structures to express either several or a particular security and privacy principle. An example is the problem frames of (Hatebur and Heizel, 2010). which provide the ability to express a problem and, through this, express an actual security principle. Another example, common in the security area, is policies. We identified the following approaches fulfilling this characteristic: [(Jutla, Bodorik and Ali, 2013), (Basso, Montecchi, Moraes, Jino and Bondavalli, 2015), (Hatebur and Heizel, 2010), (Mouheb, Talhi, Lima, Debbabo, Lang and Pourzandi, 2009), (Huang and Kirchner, 2013), (Mixia, Qiuyu, Dongmei and Hong, 2005), (Akbarzadeh and Azgomi, 2010), (Bouroulet, Devillers, Klaudel, Pelz and Pommereau, 2008), (Henry, Layer and Zaret, 2010), (Atluri and Huang, 2000)].

Each approach is assigned to one of the above characteristics. The approaches we reviewed focus either on the key feature of confidentiality to express privacy, or on introducing various other structures through which privacy is expressible. The first are grouped under the characteristic 'information flow and access control' and the latter ones under the characteristic 'general structures'. Our analysis shows that nearly half of the reviewed software architecture-oriented and business process-oriented approaches fulfil the first characteristic. They all introduce elements to model confidentiality. Some of them additionally use confidentiality mechanisms to establish privacy in a specific way [(Fernandez-Medina, Trujillo, Villaroel and Piattini, 2004), (Zhang, Hong and Liao, 2006), (Sun, Yang, Wang and Zhang, 2009), (Lai, Hong and Jeng, 2008), (Accorsi and Wonnemann, 2011), (Accorsi, Leh-mann and Lohmann. 2015), (Li, Wu and Huang, 2009), (Knorr, 2001)]. The other approaches of the first group only introduce modelling elements for confidentiality. These modelling elements are not directly for the purpose of expressing privacy [(Jürjens, 2002), (Heldal, Schlager and Bende, 2004), (Goudalo and Seret, 2008), (Simsons, 2007), (Mülle, Stackelberg and Böhm, 2011)]. The other half of the reviewed approaches utilize various other mechanisms to model privacy. The approach of [(Julta, Bodorik and Ali, 2013)], for example, introduces new structures like super containers and problem frames to express privacy. Some others use policies [(Basso, Montecchi, Moraes, Jino and Bondavalli, 2015), (Huang and Kirchner, 2013)].

5.2 Different Views

This criterion distinguishes the approaches according to their view on the model. As there are various stakeholders with different concerns to express, different views arise that fulfil the needs of a specific stakeholder. Typical examples from the field of security are the attacker view and security specialist view. The attacker view introduces model elements showing how the attacker could break into the system. The opposite side highlights the security measures in place, namely the security specialist view.

The criterion ‘different views’ divides the approaches according to the needs of their stakeholders. Common views are:

- Attacker view: models the attacker with the attacks, threats and vulnerabilities of a system, or analyses the given model for flaws in the information flow [(Jürjens, 2002), (Akbarzadeh and Azgomi, 2010), (Bouroulet, Devillers, Klaudel, Pelz and Pommereau, 2008), (Henry, Layer and Zaret, 2010), (Accorsi and Wonnemann, 2011), (Accorsi, Lehmann and Lohmann. 2015), (Li, Wu and Huang, 2009), (Atluri and Huang, 2000)].
- Requirements & Implementation view: introduces elements to express requirements pertaining to security and privacy aspects and elements, which model security and privacy solutions [(Julta, Bodorik and Ali, 2013), (Basso, Montecchi, Moraes, Jino and Bondavalli, 2015), (Heldal, Schlager and Bende, 2004), (Goudalo and Seret, 2008), (Hatebur and Heizel, 2010), (Simsons, 2007), (Mouheb, Talhi, Lima, Debbabo, Lang and Pourzandi, 2009), (Fernandez-Medina, Trujillo, Villaroel and Piattini, 2004), (Mixia, Qiuyu, Dongmei and Hong, 2005), (Zhang, Hong and Liao, 2006), (Sun, Yang, Wang and Zhang, 2009), (Lai, Hong and Jeng, 2008), (Atluri and Huang, 1996), (Atluri and Huang, 2000), (Mülle, Stackelberg and Böhm, 2011)].
- Verification view: allows users to check whether a model fulfils certain requirements by checking them against the model. This is realized, for example, with constraints, which are checked for correct implementation, or the verification of policies [(Basso, Montecchi, Moraes, Jino and Bondavalli, 2015), (Heldal, Schlager and Bende, 2004), (Fernandez-Medina, Trujillo, Villaroel and Piattini, 2004), (Huang and Kirchner, 2013), (Zhang, Hong and Liao, 2006), (Accorsi, Lehmann and Lohmann. 2015), (Li, Wu and Huang, 2009), (Knorr, 2001), (Atluri and Huang, 1996)].

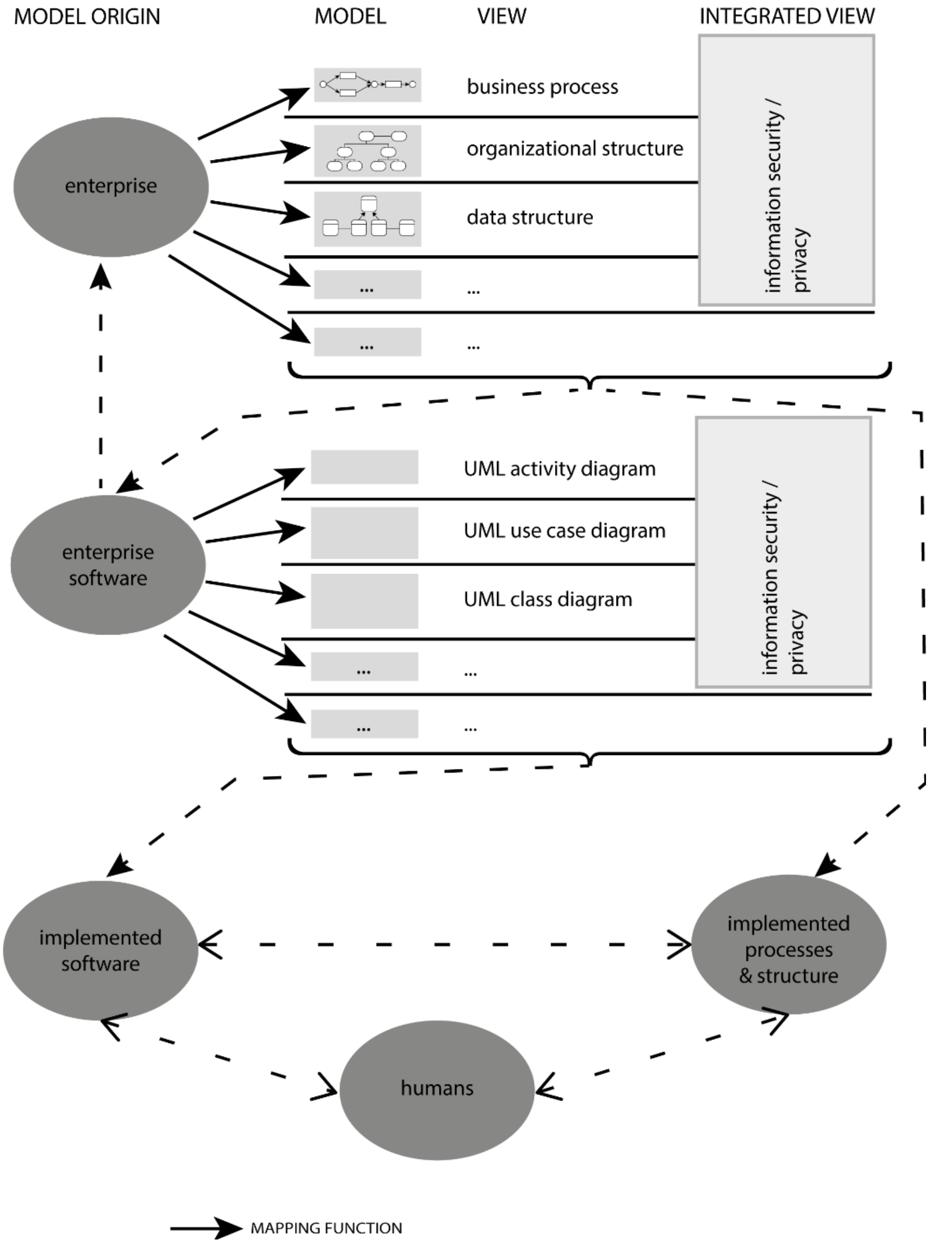
The software architecture-oriented approaches realize the ‘attacker view’ by introducing an attacker with his capabilities. We found only one approach of this type in our analysis (Jürjens, 2002). The business process-oriented side identifies flaws in the information flow, and thus privacy breaches. Both the software architecture-oriented approaches and the business process-oriented approaches are represented in the ‘requirements & implementation view’. Here, elements are introduced to express security and privacy requirements or solutions. The difference in these approaches lies in the degree of abstraction. While the business process-oriented approaches are typically on a less technical and more abstract level, the software architecture-based approaches introduce both a non-expert view and, sometimes, a more technical, expert view. In both software architecture-oriented approaches and business process-oriented approaches, we identified the intention to verify whether the implementation or model is correct with respect to certain requirements. These approaches are part of the ‘verification view’. While software architecture-oriented approaches verify the correctness of modelled solutions, business process-oriented approaches try to identify and verify security policies against a given model. In general, we recognized that, for the reviewed approaches, the software architecture-based approaches tended to model requirements or design solutions more often. They also had a stronger focus on verifying whether the model fulfils the requirements. The business process-based approaches had a stronger focus on the identification of flaws and the verification of policies.

6 Conclusion

As we have shown, there are some approaches to systematically modelling security and/or privacy aspects of organizations each from a specific perspective. However, no comprehensive approach integrates all aspects such as process, structural organization and data. Such approaches must be developed or further developed. Figure 1 illustrates the relationships between companies and enterprise software (as the origin of models), sent model types and views, as well as the implemented software, the implemented processes/structure and the people involved. The arrow shown between origin and model describes a mapping function. Dotted arrows describe influences between different original models or artefacts. Different models exist for a company (the model origin at the top of the figure). For the view Business Process Flow Models, for example, Petri Nets and/or BPMN models exists. For this purpose, we have drawn in a new integrated view, information security/privacy. This includes various other views and their models and integrates them in an appropriate manner. Appropriate links must be developed for this purpose. For example, you need to describe which organizational unit participates in a particular activity of a business process, and to determine whether the organizational unit is allowed access to the data that is also linked to the activity. In addition to this linking of existing views, an integrated view can further enhance the models (for example, by providing additional information on data protection, such as the purpose of an activity to check the purpose limitation of the data). Such an integrated view is currently not sufficiently developed for the Information Security/Privacy application case, as literature research has shown. However , approaches and concepts already exist (such as the concept modelling suites, a concrete implementation of which is, for example, the Horus Business Modeller, www.horus.biz), on the basis of which this integrated view was developed. Integrated views means that models from different views are linked together and consistency is enforced.

This integrated view describes the requirements of those responsible for the company software. These requirements of the enterprise models must be transferred into the software models to be implemented later. However, software engineers use other models (e.g. UML) to describe the requirements.

Fig. 1. Holistic Modelling Approach



Nevertheless, traceability of the requirements must be guaranteed. A systematic and, as far as possible, automatic transformation of the requirements is therefore required. This is shown in Figure 1 by the dashed line between the company models and the software. Here, it is necessary to derive an integrated view for the middle part of the illustration from the integrated view of the upper level. We therefore suggest an automated model transformation from enterprise to software modelling. Continuous modelling is a prerequisite for the traceability of the requirements. Therefore, it must be possible to transfer business requirements modelled in Petri nets to software requirements modelled in UML.

The arrow between enterprise software and the enterprise in Figure 1 shows that standard software influences enterprises as well. The arrow between the company models in their entirety and the implemented processes / structure describes the influence of modelling on subsequent execution. The connection between the software models as a whole and the implemented software is also shown by a dashed arrow. Finally, implemented software and implemented processes (which can also be partly manual) / implemented structure influence each other in terms of execution properties such as efficiency. The people involved are also affected or influence the concrete use of the software, or compliance to the processes and structures.

That there is currently no comprehensive modelling approach which covers the necessary aspects and perspectives. This should include processes as well as, for example, organizational and data structure questions. Therefore we suggest a new holistic modelling approach which includes the needed aspects and with a concept for the traceability of the requirements from business models to software architecture models. The new approach uses modelling languages and methods of existing approaches. To get a holistic view we linked them (different views and languages) and enriched them for the purpose of privacy and security modelling.

References

1. Accenture; 2017: Cost of Cybercrime Study, *https://www.accenture.com/t20170926T072837Z__w__/us-en/_acnmedia/PDF-61/Accenture-2017-CostCyberCrimeStudy.pdf, last accessed 03.05.2018.*
2. Aerts, A.T.M.; Goossenaerts, J.B.M.; Hammer, D.K.; Wortmann, J.C.; 2004: Architectures in context: on the evolution of business, application software, and ICT platform architectures. In *Information & Management, vol. 41 (6), pp. 781–794.*
3. Accorsi, R.; Wonnemann, C.; 2011: InDico: Information flow analysis of business processes for confidentiality requirements. In *Security and Trust Management, Springer, pp. 194–209.*
4. Accorsi, R.; Lehmann, A.; Lohmann, N.; 2015: Information leak detection in business process models: Theory, application, and tool support. In *Inf. Syst., vol. 47, pp. 244–257.*
5. Akbarzadeh, M.; Azgomi, M. A.; 2010: A framework for probabilistic model checking of security protocols using coloured stochastic activity networks and PDETool. In *5th International Symposium on Telecommunications (IST), pp. 210–215.*
6. Alpers, S.; Pilipchuk, R.; Oberweis, A.; Ruessner, R.; 2018: Identifying Needs for a Holistic Modelling Approach to Privacy Aspects in Enterprise Software Systems. In *Proceedings of the 4th International Conference on Information Systems Security and Privacy - Volume 1: ICISSP, pp. 74-82.*
7. Atluri, V.; Huang, W.-K.; 1996: An extended Petri net model for supporting workflows in a multilevel secure environment. In *Database Security Tenth International Conference on Database Security, Como, Italy, pp. 240–258.*
8. Atluri, V.; Huang, W.-K.; 2000: A Petri net based safety analysis of workflow authorization models. In *J. Comput. Secur., vol. 8, no. 2, 3, pp. 209–240.*
9. AXELOS; 2011: ITIL 2011 Edition, *https://www.axelos.com/best-practice-solutions/itil/what-is-itil, last accessed 20.03.2018.*
10. Basel Committee on Banking Supervision (BCBS); 2013: Third Basel Accord, *https://www.bis.org/publ/bcbs189.pdf, last accessed 20.03.2018.*
11. Basso, T.; Montecchi, L.; Moraes, R.; Jino, M.; Bondavalli, A.; 2015: Towards a UML Profile for Privacy-Aware Applications. In *IEEE International Conference on Computer and Information Technology, pp. 371-378.*
12. Bouroulet, R.; Devillers, R.; Klaudel, H.; Pelz, E.; Pommereau, F.; 2008: Modeling and analysis of security protocols using role based specifications and Petri nets. In *Applications and Theory of Petri Nets, K. M. van Hee and R. Valk, Eds. Springer Berlin Heidelberg, pp. 72–91.*
13. Crazzolara, F.; Winskel, G.; 2001: Events in security protocols. In *Proceedings of the 8th ACM conference on Computer and Communications Security, pp. 96–105.*
14. European Union; 2017: General Data Protection Regulation, http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679, *last accessed 30.04.2018.*
15. Federal Financial Supervisory Authority (BaFin); 2005: Minimum Requirements for Risk Management, *https://www.bundesbank.de/Redaktion/EN/Downloads/Tasks/Banking_supervision /PDF/minimum_requirements_for_risk_management_mindestanforderungen_an_das_risikom anagement_marisk.pdf, last accessed 30.04.2018.*
16. Federal Office for Information Security (BSI); 2015: IT Security Act, *http://www.bgbl.de/xaver/bgbl/start.xav?startbk=Bundesanzeiger_BGBl&jumpTo=bgbl115s1324.pdf, last accessed 30.04.2018.*

17. Fernandez-Medina, F.; Trujillo, J.; Villaroel, R.; Piattini, M.; 2004: Extending UML for Designing Secure Data Warehouses. In *Conceptual Modeling - ER 2004: 23rd International Conference on Conceptual Modeling, pp. 217-230.*
18. Genz, A, 2004: Datenschutz in Europa und den USA, DuD-Fachbeiträge, Deutscher Universitätsverlag, Wiesbaden.
19. German Federal Office for Information Security; 2006: IT Baseline Protection, *https://www.bsi.bund.de/EN/Topics/ITGrundschutz/ITGrundschutzCatalogues/itgrundschutzcatalogues_node.html, last accessed 20.03.2018.*
20. Goudalo, W.; Seret, D.; 2008: Toward the Engineering of Security of Information Systems (ESIS): UML and the IS Confidentiality. In: *2nd International Conference on Emerging Security Information, Systems and Technologies, pp. 248-256.*
21. Hatebur, D.; Heisel, M.: A UML Profile for Requirements Analysis of Dependable Software. In *Computer Safety, Reliability, and Security: 29th International Conference, SAFECOMP 2010, pp. 317-331*.
22. Henry, M. H.; Layer, R. M.; Zaret, D. R.; 2010: Coupled Petri nets for computer network risk analysis. In *Int. J. Crit. Infrastruct. Prot., vol. 3, no. 2, pp. 67–75.*
23. Hornung, G; Schnabel, C; 2009: Data protection in Germany I: The population census decision and the right to informational self-determination. *In Computer Law & Security Review, vol. 25(1), pp. 84–88*
24. Huang, H.; Kirchner, H.; 2013: Secure interoperation design in multi-domains environments based on colored Petri nets. *In Inf. Sci., vol. 221, pp. 591–606.*
25. Information Systems Audit and Control Association, (ISACA); 2012: COBIT, *http://www.isaca.org/COBIT/Pages/COBIT-5-Framework-product-page.aspx, last accessed 20.03.2018.*
26. International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC); 2014: ISO/IEC 27000:2014, *https://www.iso.org/standard/63411.html, last accessed 20.03.2018.*
27. Jürjens, J.; 2002: UMLsec: Extending UML for Secure Systems Development. In *Proceedings of the 5th International Conference on The Unified Modeling Language (UML '02), pp. 412-425.*
28. Jürjens, J.; 2005: Model-Based Security Engineering with UML. In *Foundations of Security Analysis and Design III: FOSAD 2004/2005 Tutorial Lectures, pp. 42-77.*
29. Jutla, D. N.; Bodorik, P.; Ali, S.; 2013: Engineering Privacy for Big Data Apps with the Unified Modeling Language. In *IEEE International Congress on Big Data, pp. 38-45.*
30. Knorr, K.; 2001: Multilevel security and information flow in Petri net workflows. In *Proceedings of the 9th International Conference on Telecommunication Systems, pp. 613–615.*
31. Labda, W.; Mehandjiev, N.; Sampaio, P.; 2013: Privacy-aware business processes modeling notation (prvbpmn) in the context of distributed mobile applications. In *Trends in Mobile Web Information Systems. Springer, pp. 120-134.*
32. Lai, H.; Hong, J.; Jeng, W.; 2008: Model E-contract update by coloured activity net. In *IEEE Asia-Pacific Services Computing Conference. APSCC '08, 2008, pp. 488–493.*
33. Leitner, M.; Miller, M.; Rinderle-Ma, S.; 2013: An Analysis and evaluation of security aspects. In *The Business Process Model and Aotation, pp. 262–267.*
34. Li, W.; Wu, R.; Huang, H.; 2009: Colored Petri nets based modeling of information flow security; *In Second International Workshop on Knowledge Discovery and Data Mining, pp. 681–684.*
35. Meland; P. H.; Gjaere, E. A.; 2012: Representing Threats. In *BPMN 2.0, pp. 542–550.*
36. Mixia, L.; Qiuyu, Z.; Dongmei, Y.; Hong, Z.; 2005: Formal security model research based on Petri-net. In *IEEE International Conference on Granular Computing, vol. 2, pp. 575–578.*

37. Mouheb, D.; Talhi, C.; Lima, V.; Debbabo, M.; Lang, L.; Pourzandi, M.; 2009: Weaving Security Aspects into UML 2.0 Design Models. In *Proceedings of the 13th Workshop on Aspect-oriented Modeling, pp. 7 – 12.*
38. Mülle, J,; Stackelberg, S. v.; Böhm, K.; 2011: Modelling and transforming security constraints in privacy-aware business processes. In *2011 IEEE International Conference on Service-Oriented Computing and Applications (SOCA), pp. 1–4.*
39. OMG; 2013: Business Process Model and Notation (BPMN) v2.0.2, *http://www.omg.org/spec/BPMN /2.0.2/PDF, last accessed 07.09.2017.*
40. OMG; 2017: Unified Modeling Language v2.5, *http://www.omg.org/spec/UML/2.5/PDF, last accessed 07.09.2017.*
41. Heldal, R.; Schlager, S.; Bende, J.; 2004: Supporting Confidentiality in UML: A Profile for the Decentralized Label Model. In *Proceeding Workshop on Critical Systems Development with UML, pp. 56-70.*
42. Reisig, W.; 2013: Understanding Petri Nets: Modeling Techniques, Analysis Methods, Case Studies, Springer, New York.
43. Shariati, M.; Bahmani, F.; Shamst, F.; 2011: Enterprise information security, a review of architectures and frameworks from interoperability perspective. *In Procedia Computer Science, vol. 3, pp. 537-543.*
44. Simons, C.; 2007: CMP: A UML Context Modeling Profile for Mobile Distributed Systems. In *40th Annual Hawaii International Conference on System Sciences, pp. 289-299.*
45. Störrle, H.; 2017: How are Conceptual Models used in Industrial Software Development? : A Descriptive Survey. *In Proceedings of the 21st International Conference on Evaluation and Assessment in Software Engineering (EASE'17), pp. 160-169.*
46. Sun, H.; Yang, J.; Wang, X.; Zhang, Y.; 2009: A verification mechanism for secured message processing in business collaboration. In *Advances in Data and Web Management, Springer, pp. 480–491.*
47. Triki, S.; Ben-Abdallah, H.; Feki, J., Harbi, N.; 2010: Modeling Conflict of interest in the Design of Secure Data Warehouses. In *KEOD 2010 - International Conference on Knowledge Engineering and Ontology Development, pp. 445-500.*
48. Wolter, C.; Meinel, C.; 2010: An approach to capture authorisation requirements. In *Business Processes. Requir. Eng., vol. 15, no. 4, pp. 359–373.*
49. Zhang, Z.-L.; Hong, F.; Liao, J.-G.; 2006: Modeling Chinese Wall Policy Using Colored Petri Nets. *In The Sixth IEEE International Conference on Computer and Information Technology, pp. 162–162.*