Achieving GDPR Compliance of BPMN Process Models

Simone Agostinelli[1], Fabrizio Maria Maggi[2],
Andrea Marrella[1], and Francesco Sapio[1]

[1] DIAG, Sapienza University of Rome, Italy
{agostinelli,marrella,sapio}@diag.uniroma1.it
[2] University of Tartu, Estonia
f.m.maggi@ut.ee

Abstract. In an increasingly digital world, where processing and exchange of personal data are key parts of everyday enterprise business processes (BPs), the right to data privacy is regulated and actively enforced in the Europe Union (EU) through the recently introduced General Data Protection Regulation (GDPR), whose aim is to protect EU citizens from privacy breaches. In this direction, GDPR is highly influencing the way organizations must approach data privacy, forcing them to rethink and upgrade their BPs in order to become GDPR compliant. For many organizations, this can be a daunting task, since little has been done so far to easily identify privacy issues in BPs. To tackle this challenge, in this paper, we provide an analysis of the main privacy constraints in GDPR and propose a set of design patterns to capturing and integrating such constraints in BP models. Using BPMN (Business Process Modeling Notation) as modeling notation, our approach allows us to achieve full transparency of privacy constraints in BPs making it possible to ensure their compliance with GDPR.

Keywords: Data Privacy · GDPR · Process Models · BPMN.

1 Introduction

Nowadays, the advances in the amount of storage and processing power have made it possible to store and process virtually all the information that might be of interest for an organization to rapidly deliver digital and physical services to their customers (e.g., the creation of a new bank account, the management of a purchase order, etc.). On the other hand, the seemingly never ending collection of customers' data by large corporations such as Google and Facebook has raised public awareness on *privacy* concerns [12].

Since May 2018, in the European Union (EU), the *right to privacy* of personal data has been tackled by the General Data Protection Regulation (GDPR) [5]. The aim of GDPR is to protect EU citizens from privacy breaches on their personal data. In summary, GDPR changes the way in which organizations handle personal information of their customers, and gives individuals enhanced rights of protection when it comes to their personal data. Since organizations that are not compliant with GDPR must face heavy fines, they are required to implement correctly the GDPR data management policies and take appropriate actions on data when requested by their customers.

To achieve compliance with GDPR, among a list of technical and non-technical challenges to be tackled [3], the regulation enforces organizations to reshape the way they approach the management of personal data stored and exchanged during the execution of their everyday business processes (BPs). Although BP modeling is well-suited for expressing stakeholder collaboration and the data flow exchanged between BP activities and participants, little has been done so far to identify potential privacy breaches in BP models [13].

Conversely, the common practice to address privacy breaches in a BP is to implement ad-hoc countermeasures during the automation stage of the BP life-cycle, when the BP model is configured by a system engineer (SE) for its execution with a dedicated BP Management System (BPMS). The SE can then implement a strategy (e.g., in the form of a piece of software) directly using the BPMS at hand, in order to deal with all potential violations of privacy constraints at run-time. However, this approach requires that the SE knows exactly where potential privacy breaches can manifest in the BP, and this information, if not explicitly documented in the BP model, may lead to a defective implementation of compensatory strategies from privacy breaches.

In this paper, we advocate that privacy should be considered as a *first-class citizen* in BP models and should be introduced *by design* and not as an afterthought. In this direction, we provide an analysis of the main privacy constraints in GDPR encountered when modeling BPs with ISO/IEC 19510:2013 BPMN (Business Process Modeling and Notation). Based on this analysis, we propose a set of design patterns to integrate privacy enhancing features in a BPMN model according to GDPR. The aim of this work is to emphasize awareness of privacy-concerns in BPs at design-time, when a proper analysis of the involved data allows a BP designer to identify (possible) violations of privacy constraints and their impact. The feasibility of our approach is illustrated using a concrete case of a phone company.

The rest of the paper is organized as follows. Section 2 introduces a case in which privacy aspects of a BP for acquiring a new customer by a phone company need to be modeled. Section 3 introduces the main constraints of GDPR considered in the paper. Section 4 presents a set of design patterns to capturing and integrating GDPR constraints in BP models. Finally, Section 5 illustrates the relevant literature related to privacy in BPs and concludes the paper.

2 The Case of a Phone Company

With the increase of systems able to collect data automatically, privacy has been at the center of many discussions between designers who want to use such data to provide services to the users, and these last ones who want to get the services by sharing as little information as possible. In fact, users care about any data that can identify them, and this has many consequences in the corpus of laws in all countries, although different countries have different boundaries of what can be considered private information.

Let us take as an example a phone company in the process of acquiring a new customer. The phone company requests the new client's data (e.g., name, surname, address, etc.). Once the client has provided this data, the phone company goes through a verification process to determine if the data given by the new customer is correct. If not, a clear up procedure starts and the process ends. The next steps involve asking the future customer if she wants to *port* her old phone number into the new phone plan she is about to subscribe. If the answer is positive, then the phone company asks the new client the old number, and the portability procedure to the *previous phone company*. In case the procedure can not be completed or the answer is negative, the process is interrupted. Otherwise, the customer signs the contract, which only describes how the phone company will provide the service, but which does not provide any information on how the phone company will use the data of the client. After this, the phone company in parallel stores the data of the new client, requests the payment to the client, and, once the payment has been received, sends the SIM card to the client. Once these activities have been completed, the company can activate the SIM card and successfully conclude the procedure of acquiring the new customer. If the procedure takes for some reasons more than 30 days to complete, then the process is interrupted.

The BPMN model representing the scenario described above is shown in Fig. 1. It is worth noting that the procedure does not yet take into account the potential risk to get a data breach and does not provide mechanisms to protect the customer's privacy.

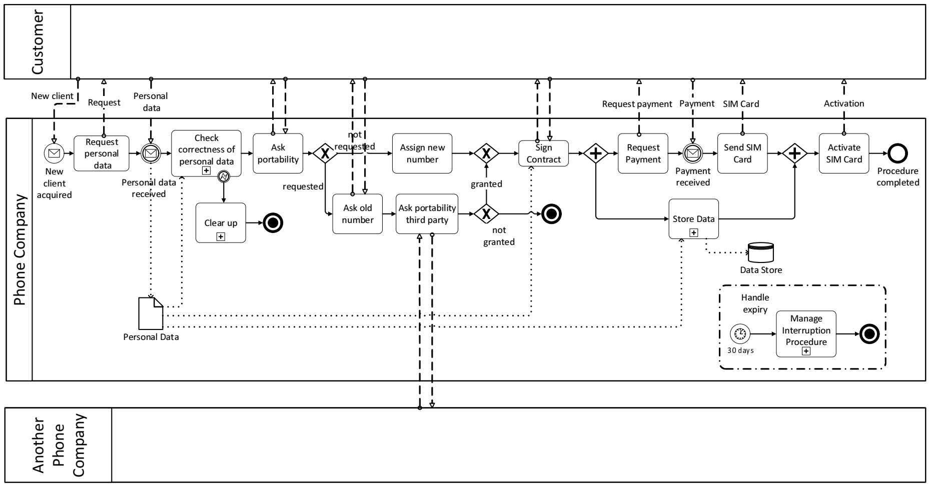

Fig. 1. BPMN model of the case of the phone company

3 Background on GDPR

GDPR has introduced changes to the privacy and data protection regulation, thus having significant consequences for who needs to design BPs. GDPR requires *privacy-by-design*, which means that data protection is not an addition to the process, but rather integral part of it, and the process should comply to GDPR since the design stage. Therefore, already at this stage, a BP designer needs to take into consideration privacy and data protection issues.

Entities involved. In order to identify who is responsible of what in a BP where data is handled, GDPR defines four entities:

- *Data Subject*: is the person the data is about.
- *Data Controller*: is the entity that collects and stores data from the Data Subject and that determines the purposes of processing such data.
- *Data Processor*: is the entity that processes data from the Data Subject on the behalf of the Data Controller.
- *Data Protection Officer (DPO)*: is the entity that performs systematic monitoring on the Data Controller and Data Processor to ensure that they comply with the GDPR constraints on the data collected from the Data Subject.

Personal Data. In the context of GDPR, *Personal data* is defined as any information related to a person (Data Subject) who can be identified, directly or indirectly, through that information (e.g., a name, an identification number, location data, online identifiers, etc.). Therefore, online identifiers, including IP address and cookies, will now be regarded as personal data if they can be linked back to the Data Subject. GDPR distinguishes three types of personal data,[3] each with a different level of protection:

- *Personal Data*: any piece of information that can identify a person.
- *Sensible Data*: is a special type of *Personal Data* that requires a higher level of security, i.e., health, genetic, physical, physiological, mental, economic, cultural, social identity and biometric data.
- *Criminal Records*: is a subset of *Sensible Data* including information to identify past crimes committed by the Data Subject.

Obligations of the Data Controller. This paper focuses on the obligations of the Data Controller. This implies a list of constraints that must be fulfilled by the Data Controller to be complaint with GDPR. These obligations are:

- *Data Breach*: in case of a data breach, the Data Controller has to communicate it within 72 hours to the National Authority as well as to the Data Subject. This constraint is not subject to any *de minimis* standard, thus any data breach, it does not matter how small, needs to be always communicated in a simple way along with the actions that will be performed to limit the damage. The only exception is the case in which the stolen data is not usable (e.g., encrypted). However, also in this case, the National Authority can force the Data Controller to communicate the breach to the Data Subject.

[3] The only exception is National Security Data that does not follow GDPR regulation, but is left to the jurisdiction of each State.

- *Consent to Use the Data*: when retrieving personal data, the Data Controller needs to ask the Data Subject for consent and to provide the Data Subject with information about the intentions on how to use and/or process the data.
- *Right to Access and Rectify*: at any moment, the Data Subject has the right to *access* and *rectify* the personal data associated to her. As a result, the Data Controller has the obligation to satisfy these requests.
- *Right of Portability*: at any moment, the Data Subject has the right to ask for the portability of the data associated to her to third parties and the Data Controller has the obligation to satisfy this request.
- *Right to Withdraw*: at any moment, the Data Subject can withdraw the consent to use the data associated to her and the Data Controller has to stop using such data.
- *The Right to be Forgotten*: if the Data Subject wants her data to be deleted, the Data Controller has the obligation to satisfy this request.

4 Implementing GDPR-aware Patterns in BPMN

In this section, we introduce a list of seven privacy patterns for BPMN, which represent effective design-time solutions to tackle GDPR constraints in BP models. Notably, we developed such patterns in a way that no additional BPMN symbol is required to integrate them into a non-GDPR compliant BP model.

4.1 Data Breach

In case of a Data Breach, the Data Controller has to retrieve the breached data. From this data, the Data Controller needs to extract a list of Data Subjects who had their data breached. Then, in parallel, the Data Controller needs to limit the data loss and send a notification to the National Authority. For each breached Data Subject, the Data Controller evaluates if the stolen data is usable or not. If not, and if the Data Controller is proven to manage data using high security standards, this is communicated to the National Authority who decides whether the breach should be communicated to the Data Subject or not. Otherwise, the Data Controller needs to notify the Data Subject directly. The design pattern in Fig. 2 implements the privacy constraint *Data Breach.* It is worthwhile noting that, during any process involving personal data, a data breach can occur, and the Data Controller must promptly handle the problem within 72 hours.

In the example of the phone company, a data breach can happen at any time after the personal data has been acquired. Thus, implementing the *Data Breach* pattern can help the process to be reactive in case of data breach, so to properly provide a recovery procedure and communicate the data breach to both the Data Subject and the National Authority. Notice that if the 72 hours limit is not respected and the Data Controller is not able to provide a reasonable justification, the penalties amount to 20 millions, or 4% of the company's global revenue, whichever is higher.

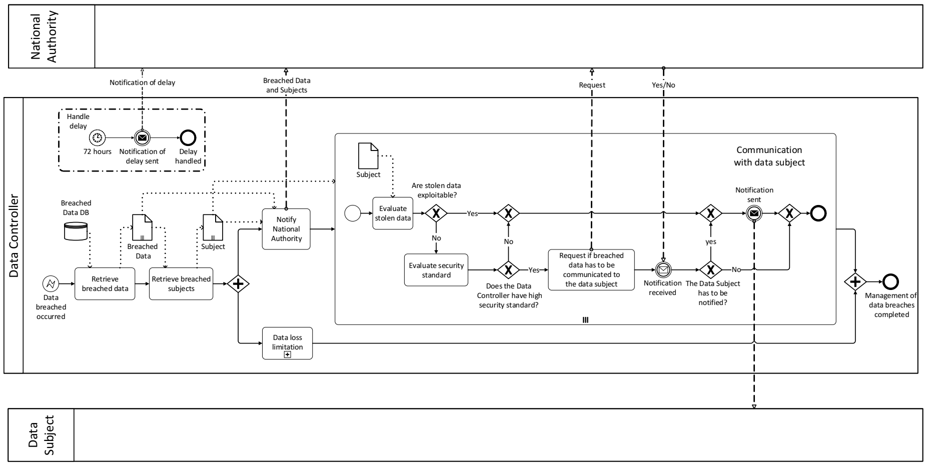
National Authority
Notification of delay
Breached Data and Subjects
Request
Yes/No
Data Controller
Handle delay
72 hours
Notification of delay sent
Delay handled
Breached Data DB
Breached Data
Subject
Data breached occurred
Retrieve breached data
Retrieve breached subjects
Notify National Authority
Data loss limitation
Communication with data subject
Subject
Are stolen data exploitable?
Evaluate stolen data
Yes
No
Evaluate security standard
No
Yes
Does the Data Controller have high security standard?
Request if breached data has to be communicated to the data subject
Notification received
The Data Subject has to be notified?
yes
No
Notification sent
Management of data breaches completed
Data Subject

Fig. 2. BPMN model for pattern *Data Breach*

4.2 Consent to Use the Data

Before retrieving any kind of personal data from the Data Subject, the Data Controller has to ask the Data Subject for consent. In particular, the Data Controller needs to collect a list of aspects the Data Subject should be aware of before giving her data to the Data Controller. This list should contain:

- in case the data has not been directly obtained from the Data Subject, from which source the personal data originates;
- the existence of the right to lodge a complaint to a supervisory authority;
- the existence of the right to withdraw the consent at any time;
- the existence of the right to data portability;
- the existence of the right to delete the personal data;
- the existence of the right to access the personal data;
- the existence of the right to rectify the personal data;
- the period for which the personal data will be stored, or if this is not possible, the criteria used to determine this period;
- the existence of any profiling and meaningful information about the envisaged consequences of processing the personal data;
- if the personal data can be transferred internationally;
- who are the recipients or categories of recipients of the personal data;
- which are the interests pursued by the Data Controller or by third parties;
- the legal basis of the processing;
- the purposes for which the personal data will be processed;
- the identity and the contact details of the Data Controller and of the DPO.

Then, the consent to use the data is requested to the Data Subject. If the consent is given, the data is collected. The design pattern in Fig. 3 implements the privacy constraint *Consent to Use the Data*.

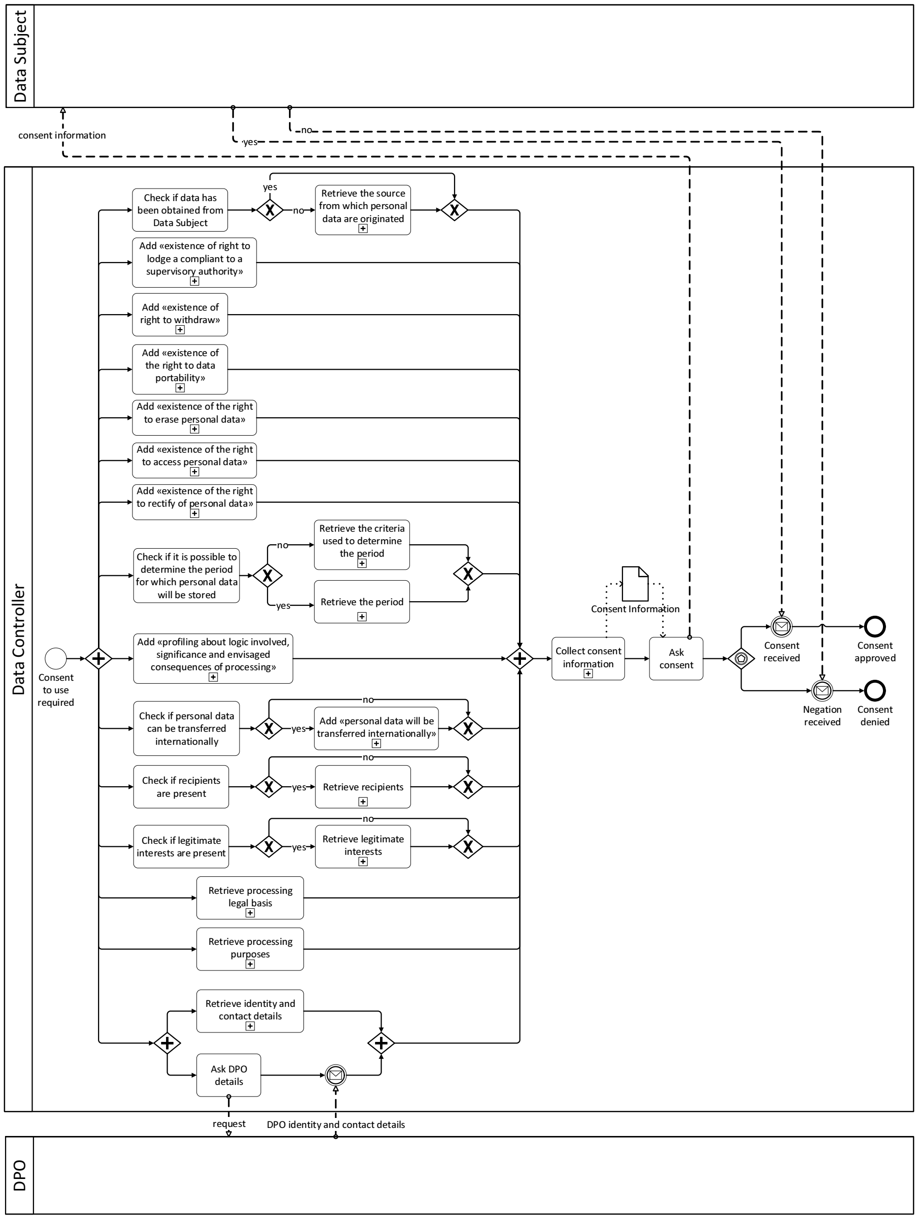
Data Subject
consent information
yes
no
Data Controller
Consent to use required
Check if data has been obtained from Data Subject
yes
no
Retrieve the source from which personal data are originated
Add «existence of right to lodge a compliant to a supervisory authority»
Add «existence of right to withdraw»
Add «existence of the right to data portability»
Add «existence of the right to erase personal data»
Add «existence of the right to access personal data»
Add «existence of the right to rectify of personal data»
Check if it is possible to determine the period for which personal data will be stored
no
Retrieve the criteria used to determine the period
yes
Retrieve the period
Add «profiling about logic involved, significance and envisaged consequences of processing»
Check if personal data can be transferred internationally
no
yes
Add «personal data will be transferred internationally»
Check if recipients are present
no
yes
Retrieve recipients
Check if legitimate interests are present
no
yes
Retrieve legitimate interests
Retrieve processing legal basis
Retrieve processing purposes
Retrieve identity and contact details
Ask DPO details
Consent Information
Collect consent information
Ask consent
Consent received
Consent approved
Negation received
Consent denied
request
DPO identity and contact details
DPO

Fig. 3. BPMN model for pattern *Consent to Use the Data*

In the example of the phone company, this pattern can be added as a subprocess just before asking for the actual data to the potential new customer, at the start of the process. This guarantees that the company is transparent with the customer and asks for the explicit consent of any possible usage of the data.

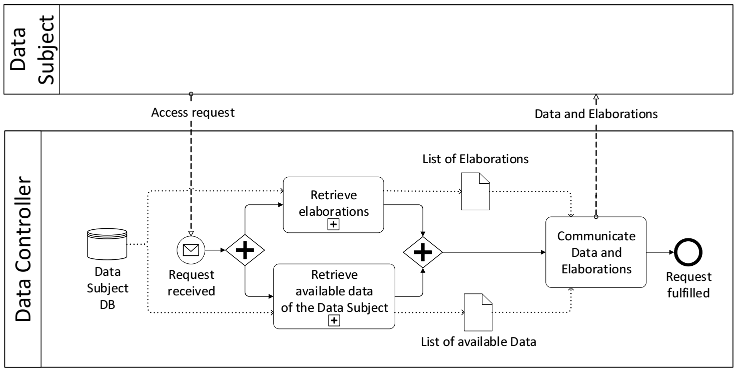

Fig. 4. BPMN model for pattern *Right to Access*

4.3 Right to Access

When the Data Subject sends a request availing the right to access, the Data Controller has to *(i)* retrieve all the data associated with the Data Subject, and *(ii)* retrieve any processing on the data that has been made. Then, they are both sent to the Data Subject. The design pattern in Fig. 4 implements the privacy constraint *Right to Access*.

In the example of the phone company, this pattern can be implemented as an asynchronous request from the Data Subject that can be received at any point in time after that any personal data has been retained. In BPMN, this pattern can be used as an event sub-process that handles the request.[4] In the example of the phone company, the customer can request to access her personal data even before the process is completed (potentially even before the customer signs the contract), and the phone company has to handle this request by providing any personal data it possesses.

4.4 Right of Portability

When the Data Subject sends a request availing the right of portability, she needs to specify the third party at hand. The third party contacts the Data Controller which has to *(i)* retrieve all the data associated with the Data Subject, and *(ii)* retrieve any processing on the data that has been made. Then, they are both sent to the third party. Finally, the third party communicates to the Data Subject that the portability happened successfully. The design pattern in Fig. 5 implements the privacy constraint *Right of Portability*.

In the example of the phone company, the company needs to have a procedure to handle portability when requested by a third party company. However, in the process of acquiring a new client, even though the user requests the portability,

[4] Event sub-processes are used in BPMN to capture exceptions (and define recovery procedures) that may affect an entire BP.

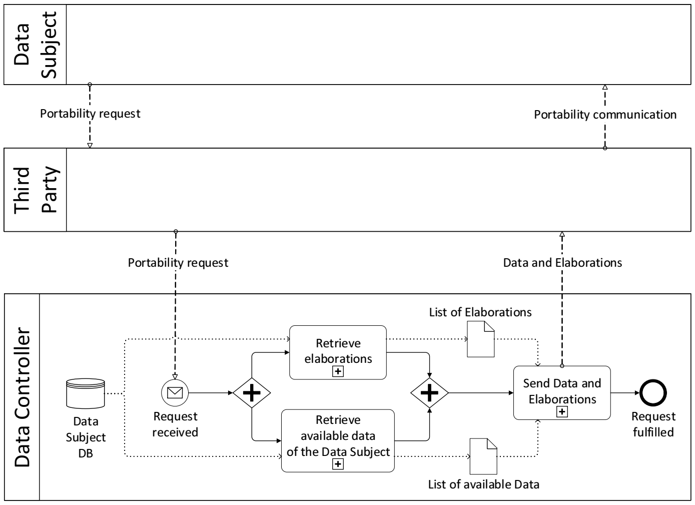

Fig. 5. BPMN model for pattern *Right of Portability*

Another Phone Company (and not the phone company of the case study) should be able to implement this pattern.

4.5 Right to Withdraw

When the Data Subject sends a request availing the right to withdraw, the Data Controller has to stop using the data associated to the Data Subject, and communicate back to the Data Subject that her data is not used anymore. The design pattern in Fig. 6 implements the privacy constraint *Right to Withdraw*.

In the example of the phone company, this asynchronous request from the client can happen at any time, thus the phone company might implement this pattern in BPMN as an event sub-process. If, at any time during the procedure of acquiring a new customer, the customer withdraws the consent to use the data, the phone company has to evaluate if that data is needed to continue the process. If this is the case, the process will terminate, since the phone company is not able to complete the procedure.

4.6 Right to Rectify

When the Data Subject sends a request availing the right to rectify, the Data Controller has to rectify the data as requested by the Data Subject, and communicate back to the Data Subject that her data has been rectified. The design pattern in Fig. 7 implements the privacy constraint *Right to Rectify*.

In the example of the phone company, the customer should be able to rectify the data at any time. For instance, if before signing the contract the customer changes address, or simply notices incorrect information, she should be able to

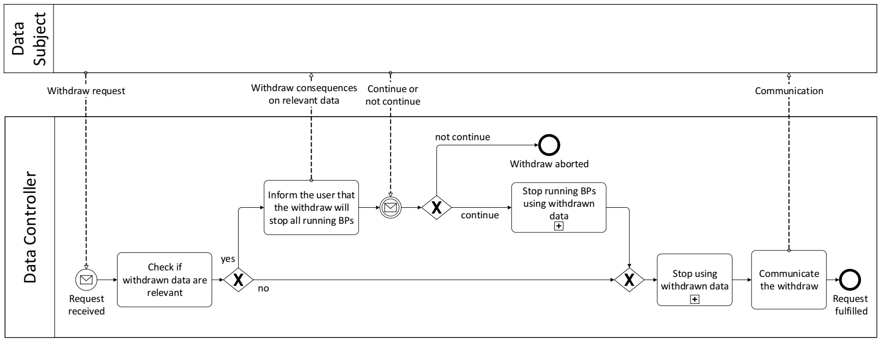
Data Subject
Withdraw request
Withdraw consequences on relevant data
Continue or not continue
Communication
Data Controller
not continue
Withdraw aborted
Inform the user that the withdraw will stop all running BPs
continue
Stop running BPs using withdrawn data
yes
Request received
Check if withdrawn data are relevant
no
Stop using withdrawn data
Communicate the withdraw
Request fulfilled

Fig. 6. BPMN model for pattern *Right to Withdraw*

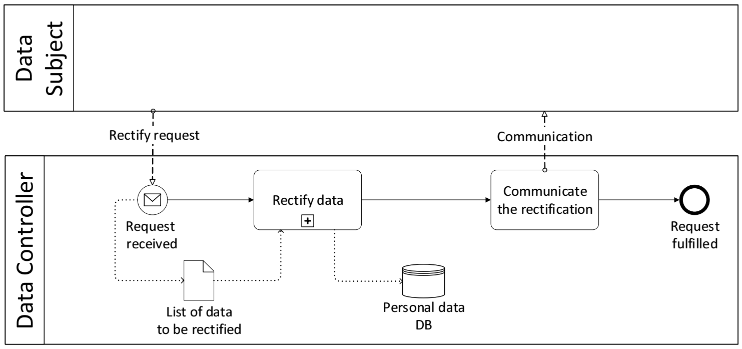

Fig. 7. BPMN model for pattern *Right to Rectify*

rectify such information. This asynchronous request could be satisfied in BPMN using an event sub-process.

4.7 Right to be Forgotten

When the Data Subject sends a request availing the right to be forgotten, the Data Controller has to retrieve the data related to the request and check if this data is relevant. If not, the Data Controller eliminates such data and communicates this to the Data Subject. Otherwise, the Data Controller communicates to the Data Subject why the data is relevant. The design pattern in Fig. 8 implements the privacy constraint *Right to be Forgotten.* This is, once again, an asynchronous request from the Data Subject, which in BPMN can be implemented as an event sub-process.

In the example of the phone company, this pattern can be implemented during the process of acquiring a new customer even though the request will be for sure rejected (since all the data requested from the clients is necessary at this stage). This is needed to provide the customer with an understanding of why the data is relevant within the process.

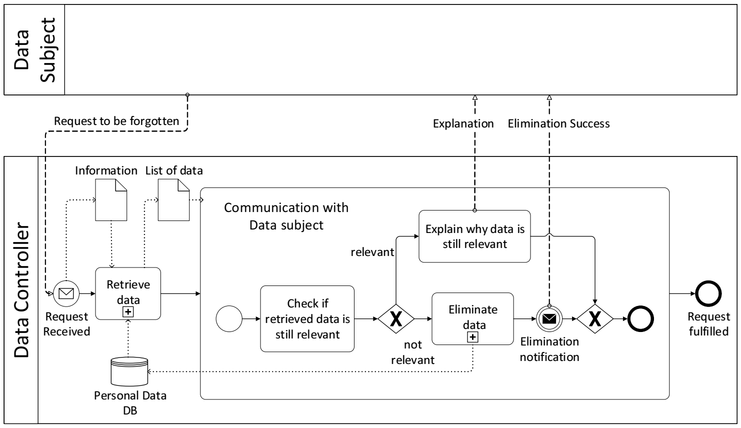

Fig. 8. BPMN model for pattern *Right to be Forgotten*

5 Related Work and Conclusion

In the literature, there are several studies where BPMN is extended towards security and privacy aspects. BPMN security extensions for healthcare processes are presented in [15,17]. Menzel et al. introduce security elements for BPMN to evaluate the trustworthiness of participants based on a rating of enterprise assets and to express security intentions such as confidentiality or integrity on an abstract level [11]. In [6], BPMN is enriched with information assurance and security modeling capabilities. In [1], BPMN is aligned to the domain model of security risk management. In [13], Privacy Enhancing Technologies (PETs) are applied to enforce privacy requirements and support the analysis of private data leakage. In [16], Salnitri et al. propose the SecBPMN-Q query language for representing security policies and a query engine that enables checking SecBPMN-Q policies against SecBPMN-ml specifications.

Some works are specifically related to the definition of extensions of BPMN to represent cyber security requirements [7,10]. In [9], the authors investigate a new approach to modeling security and propose a solution to include all concepts potentially modelable in BPMN related to cyber security. In [2], the BPMN choreography models are used to detail message exchange and identity contract negotiation. In [4], BPMN is extended with access control, separation of duty, binding of duty and need to know principles. Similarly to [4], in [8] privacy concerns are captured by annotating the BPMN model with access control, separation of tasks, binding of tasks, user consent and necessity to know icons.

Differently from the above studies, our work is focused on GDPR. Specifically, we have provided an analysis of the main privacy constraints in GDPR and a set of design patterns to capturing and integrating such constraints in BP models represented in BPMN. Recent works concerning GDPR have been also presented in [14,18]. In [14], the authors propose a method to support the design of GDPR compliant systems, based on a socio-technical approach composed of a modeling language and a reasoning framework. In [18], the authors present a model of GDPR that provides a visual overview of the associations between entities defined in the legislation and their constraints. If compared with [14,18], the originality of our approach lies in considering awareness of GDPR constraints at design-time, during BP modeling, and not as a run-time issue.

Our work can be extended in many aspects. For example, an extensive validation of the patterns against larger case studies is crucial to test the effectiveness of the overall approach. Nonetheless, we consider this work as an important first step towards a thorough understanding of how to build GDPR-aware processes.

References

1. Altuhhova, O., Matulevicius, R., Ahmed, N.: An Extension of Business Process Model and Notation for Security Risk Management. IJISMD **4**(4) (2013)
2. Ayed, G.B., Ghernaouti-Helie, S.: Processes View Modeling of Identity-related Privacy Business Interoperability: Considering User-Supremacy Federated Identity Technical Model and Identity Contract Negotiation. In: ASONAM'12 (2012)
3. Basin, D., Debois, S., Hildebrandt, T.: On purpose and by necessity: compliance under the GDPR. Financial Cryptography and Data Security **18** (2018)
4. Brucker, A.D.: Integrating Security Aspects into Business Process Models. Information Technology **55**(6) (2013)
5. Carey, P.: Data protection: a practical guide to UK and EU law. Oxford University Press, Inc. (2018)
6. Cherdantseva, Y., Hilton, J., Rana, O.F.: Towards SecureBPMN - Aligning BPMN with the Information Assurance and Security Domain. In: BPMN'12 (2012)
7. Chergui, M.E., Benslimane, S.M.: A Valid BPMN Extension for Supporting Security Requirements Based on Cyber Security Ontology. In: MEDI'18 (2018)
8. Labda, W., Mehandjiev, N., Sampaio, P.: Modeling of privacy-aware business processes in BPMN to protect personal data. In: SAC'14. pp. 1399–1405 (2014)
9. Maines, C.L., Zhou, B., Tang, S., Shi, Q.: Adding a Third Dimension to BPMN as a Means of Representing Cyber Security Requirements. In: DeSE'16 (2016)
10. Maines, C.L., Llewellyn-Jones, D., Tang, S., Zhou, B.: A Cyber Security Ontology for BPMN-Security Extensions. In: CIT'15 (2015)
11. Menzel, M., Thomas, I., Meinel, C.: Security Requirements Specification in Service-Oriented Business Process Management. In: ARES'09 (2009)
12. Petersen, S.A., Mannhardt, F., Oliveira, M., Torvatn, H.: A Framework to Navigate the Privacy Trade-offs for Human-Centred Manufacturing. In: PRO-VE'18 (2018)
13. Pullonen, P., Matulevicius, R., Bogdanov, D.: PE-BPMN: Privacy-Enhanced Business Process Model and Notation. In: BPM'17 (2017)
14. Robol, M., Salnitri, M., Giorgini, P.: Toward GDPR-Compliant Socio-Technical Systems: Modeling Language and Reasoning Framework. In: POEM'17 (2017)
15. Rodríguez, A., Fernández-Medina, E., Piattini, M.: A BPMN Extension for the Modeling of Security Requirements in Business Processes. IEICE Tr. **90**(4) (2007)
16. Salnitri, M., Dalpiaz, F., Giorgini, P.: Designing secure business processes with SecBPMN. Software and System Modeling **16**(3), 737–757 (2017)
17. Sang, K.S., Zhou, B.: BPMN Security Extensions for Healthcare Process. In: CIT'15 (2015)
18. Tom, J., Sing, E., Matulevicius, R.: Conceptual Representation of the GDPR: Model and Application Directions. In: BIR'18 (2018)