information

MDPI

Article

A Framework for Systematic Refinement of Trustworthiness Requirements

Nazila Gol Mohammadi * and Maritta Heisel

paluno—The Ruhr Institute for Software Technology, University of Duisburg-Essen, Duisburg 47057, Germany; maritta.heisel@uni-due.de

* Correspondence: nazila.golmohammadi@uni-due.de; Tel.: +49-203-379-1929

Academic Editors: Steven Furnell, Sokratis K. Katsikas and Costas Lambrinoudakis
Received: 16 December 2016; Accepted: 15 April 2017; Published: 20 April 2017

Abstract: The trustworthiness of systems that support complex collaborative business processes is an emergent property. In order to address users' trust concerns, trustworthiness requirements of software systems must be elicited and satisfied. The aim of this paper is to address the gap that exists between end-users' trust concerns and the lack of implementation of proper trustworthiness requirements. New technologies like cloud computing bring new capabilities for hosting and offering complex collaborative business operations. However, these advances might bring undesirable side effects, e.g., introducing new vulnerabilities and threats caused by collaboration and data exchange over the Internet. Hence, users become more concerned about trust. Trust is subjective; trustworthiness requirements for addressing trust concerns are difficult to elicit, especially if there are different parties involved in the business process. We propose a user-centered trustworthiness requirement analysis and modeling framework. We integrate the subjective trust concerns into goal models and embed them into business process models as objective trustworthiness requirements. Business process model and notation is extended to enable modeling trustworthiness requirements. This paper focuses on the challenges of elicitation, refinement and modeling trustworthiness requirements. An application example from the healthcare domain is used to demonstrate our approach.

Keywords: trust; trustworthiness requirements; requirements engineering; goal modeling; business process modeling

1. Introduction

Advances in Information and Communication Technology (ICT) facilitate the automation of business processes and consequently increase organizations' efficiency. For instance, cloud, social and mobile computing have been an important enabler for developing business information systems that support nowadays' complex businesses. These new technologies bring new capabilities for hosting and offering highly dynamic and collaborative business processes, e.g., healthcare services via the Internet in the medical domain. However, using these new ICTs can also bring undesirable side effects. For instance, using cloud computing may introduce new vulnerabilities and threats caused by collaboration and data exchange over the Internet. The trustworthiness of business information systems that support collaborative business processes is a key factor for promoting such collaboration and consequently the adoption of these systems. The consumers of business processes (either organizations or individuals) often hesitate in placing their trust in such technologies. Since trust is the prerequisite for performing many kinds of transactions and collaborations, users' concerns about the trustworthiness of these business processes, their involved apps, systems and platforms, slow down their adoption [1].

Trustworthiness requirements must be assured, in order to meet users' trust concerns. To support users' confidence (leading to business services adoption), the right mechanisms should be put into place. Trustworthiness requirements should be in accordance with the end-users' trust concerns.

Furthermore, business processes and their involved software systems and services need to be made trustworthy to mitigate the risks of engaging those systems.

The peculiarity about trust is, first, that it is subjective for different groups of end-users and, second, that it is achieved by satisfying a set of other qualities or properties. Consider, for example, a healthcare system that involves not only the elderly person who is monitored at home, but also the hospital, doctors, ambulance service, insurances, etc. For the elderly person, the system is trustworthy if it provides a certain degree of reliability, availability and usability. For a hospital, in contrast, it is trustworthy if the privacy of sensitive data is guaranteed. Additionally, for the doctors, the system is trustworthy if the correctness of raised alarms is ensured. Therefore, for being trustworthy, business information systems must fulfill a variety of qualities and properties. Software systems that provide support to different stakeholders should fulfill a variety of qualities and properties for being trustworthy, depending on the application and the domain. The traditional development methodologies do not respect users' trust concerns in dynamic, heterogeneous and distributed settings. Recently, innovative technologies, like trustworthiness-by-design methodologies [2], are attracting researchers' attention. Requirements engineering is a critical activity in such "by-design" methodologies. However, there is only a small set of well-accepted requirement refinement methods and complementary decision support (supporting design decisions), which can be applied in a systematic way for considering trustworthiness [3,4]. We believe that trustworthiness of business systems is strongly dependent on their development processes, especially the elaboration of trustworthiness requirements during the requirement engineering phase.

To refine and elaborate the requirements along with design artifacts in addressing trust concerns, we propose a framework to refine and analyze trustworthiness requirements in a systematic and iterative way. Trust concerns are identified and addressed in the goal models by trustworthiness goals. Trustworthiness requirements are refined in goal models iteratively in combination with the business process models defined for satisfying the goals. In this way, it is ensured that trustworthiness requirements will not be violated or ignored, while developing or implementing the activities, resources and data objects involved in the business processes.

Business process models are frequently used in software development for understanding the behavior of the users, their requirements and for the assignment of requirements to particular well-defined business process elements. In business processes, resources are either human or non-human assets, e.g., software, apps or IT devices [5]. Non-human assets can provide either fully-automated or semi-automated support to the activity performers. Since end-users rely on these technical resources when performing their activities, trustworthiness properties of these technical resources play a major role in gaining the trust of end-users (e.g., the reliability of the system that deals with monitoring the vital signs of a patient). There are specific conditions that must be defined concerning human resources that contribute as well to trustworthiness, e.g., people's skills and expertise when performing particular tasks. In addition to trustworthiness requirements on resource management, the usage of digital documents and data plays a central role in trustworthiness. For instance, in order to respect privacy regulations, digital documents have to be protected from unauthorized use (e.g., being shared on public networks). This clearly demands the consideration of trustworthiness properties and, hence, the specification of trustworthiness requirements on data objects by defining usage rules, as well as the respective mechanisms for enforcing the usage of such rules. Consequently, trustworthiness should be considered in the management of both human and non-human resources in all stages of the business process life-cycle: design, modeling, implementation, execution, monitoring and analysis.

In the state of the art, issues related to security have been widely studied. Since trustworthiness covers a broader spectrum of properties rather than just security, there is a gap in research when addressing socio-economical factors of trustworthiness.

We propose a conceptual model and a framework for systematic elicitation, refinement and modeling trustworthiness requirements in a user-centered manner. The paper aims at bringing

together trustworthiness requirements analysis with regard to trust concerns and thereafter building trustworthiness properties into underlying systems for performing business processes. The major aim is closing the existing gap between end-users' trust concerns and the lack of implementation of the appropriate trustworthiness properties in software systems. Our objectives are to analyze and specify trustworthiness requirements in the business process models to support the process designer and tool developers in fulfilling trustworthiness requirements and a later evaluation of them. We use i* [6] for goal-modeling and the Business Process Model and Notation (BPMN) [7] for modeling business processes. We also focus on specifying trustworthiness requirements starting from the business processes level by providing modeling capabilities to understand and express trustworthiness requirements. The main challenges that we discovered based on an analysis of the state of the art are a lack of concepts relevant for trustworthiness (e.g., delegations) and a lack of inter-model consistency checks between BPMN and i* models. Goal models combined with business process models specify how business processes fulfill the trustworthiness goals.

Our approach is beneficial for the decision support during run-time adaptation, as well. In an uncertain and changing environment, business processes are continuously optimized, e.g., via service substitution. To respect the overall trustworthiness level, quality trade-offs should respect trustworthiness requirements. The business process models enhanced with trustworthiness properties are useful information during the run-time, as well. Tools and services developers are supported through detailed trustworthiness requirements for the software and services to be built. We also believe that once trustworthiness requirements have been considered and documented in business process models, they will not be ignored during design-time.

This paper is an expanded version of [8]. The remainder of the paper is structured as follows: In Section 2, we explain the fundamentals of our framework. Section 3 presents our framework for combining goal models and business process modeling to support eliciting and refinement of trustworthiness requirements and embedding them in business process models. The classification of trustworthiness requirements, which can be expressed in the business process model, is also described. The method for elicitation and refinement of trustworthiness requirements is presented in Section 4. We demonstrate the application of our framework in a case study inspired from the EU project OPTET (http://www.optet.eu/) in Section 5. In Section 6, the benefits of applying our method are presented. In Section 7, related work is discussed. Section 8 provides an initial validation and evaluation plan. Finally, we conclude our work and sketch future work in Section 9.

2. Background and Fundamentals

In this section, we briefly introduce the fundamental techniques and concepts for the framework that is described in Section 3.

2.1. Trust and Trustworthiness

Trust is defined as a "bet" about the future contingent actions of a system [9]. The components of this definition are belief and commitment. There is a belief that placing trust in software or a system will lead to a good outcome. Then, the user commits by placing trust in the software or system and taking an action like using it. This means when a user decides to use a service, e.g., a healthcare service on the web, she/he is confident that it will meet her/his expectations. Trust is subjective. For instance, in a home monitoring system, an insurance company requires confidence about its business-critical data, whereas an elderly person using the system may be more concerned about usability. Once these concerns are elicited, they manifest as trustworthiness requirements.

Trustworthiness properties are the qualities of the system that potentially influence trust in a positive way. The term trustworthiness is not used consistently in the literature. Trustworthiness has sometimes been used as a synonym for security and sometimes for dependability. However, security is not the only aspect of trustworthiness. Some approaches merely focus on single trustworthiness characteristics, e.g., security or privacy. However, trustworthiness is rather a broad-spectrum term

with notions including reliability, security, performance and usability [10]. Trustworthiness is domain and application dependent. For instance, in healthcare applications, the set of properties that have primarily been considered consists of availability, confidentiality, integrity, maintainability, reliability and safety, but also performance and timeliness.

2.2. Business Process Modeling Using BPMN

A business process is a specific ordering of activities across time and place, with a start, an end and clearly-defined inputs and outputs. A business process model is the representation of the activities, documents, people and all of the elements involved in a business process, as well as the execution constraints between them [11]. By using business process modeling, different information can be captured such as organizational, functional, informational, behavioral and context information. The organizational information focuses on the actors and their activities. The functional information describes the process element activity that is being performed during a business process execution. A resource can either be a human resource or a technical resource, such as tools, or a service used in performing an activity, or informational resources, such as data. The business process models also represent how the informational resources are manipulated in a process. The behavioral information includes the time aspects of activities by focusing on when activities are performed and when they are sequenced. We can show control flow and data flow in business process models.

BPMN [7] is a standard for modeling business processes, which is broadly extended and used widely in both industry and research. The most important BPMN elements are shown in Figure 8.

*2.3. Goal Modeling Using i**

In requirements engineering, goal modeling approaches have gained considerable attention in varying contexts. These approaches aim at capturing the rationale of the software system development. A goal model defines organization goals and the tasks necessary to achieve these goals. Thus, goal models relate the high-level goals of an organization to low-level system requirements. Goals can be classified into two different categories: hard goals and soft goals. Soft goals can be achieved at different levels of satisfaction, which means that there is no clear-cut definition for their satisfaction [6]; whereas, for hard goals, the condition for judging whether a goal is satisfied is clearly defined.

Hard goals may refer to the functional properties of the system behavior, whereas soft goals may represent quality preferences. There exists a number of different goal modeling languages used in requirements engineering.

The i* notation was developed with the purpose of modeling and reasoning within an organizational environment and its information systems [6]. It consists of two main models, a Strategic Dependency Model (SDM) and a Strategic Rationale Model (SRM). The SDM (cf. Figure 7) is used to express strategic relationships among different actors in an organizational context. The SRM (cf. Figure 9) captures both an internal view of each actor and external relations among actors. The main concepts used in i* models are actors, goals, tasks, resources and soft goals. An actor is a role who carries out a task to achieve a certain goal. A resource is an object that is needed to complete a goal or to perform some task. The following dependencies can be defined in i*: goal, soft goal, task or resource dependencies (cf. Figure 7). For the internal view of an actor in an SRM, the links are as follows: means-ends, task decomposition and contribution (cf. Figure 9).

2.4. Choice of i and BPMN as Modeling Languages in Our Approach*

We analyzed different goal modeling languages with respect to their ability to assist business process models in addressing trustworthiness. It is important that the chosen languages are able to model the concepts given in our conceptual model (cf. Figure 3). Every goal modeling language that supports these concepts can be used. For instance, goal modeling languages can be used where goals are assigned to actors (or so-called agents). The same condition, namely that the concepts of our conceptual model can be mapped to it, holds for business process modeling languages.

As a goal modeling language, we have selected i*. However, also other languages, such as Knowledge Acquisition in autOmated Specification (KAOS) [12,13], could be used. i* is applied when business-level analysis is the starting point of the modeling, while KAOS is widely used in goal-oriented requirements engineering for IT systems. One of the major reasons to use i* instead of KAOS is the possibility of analyzing trade-off decisions based on dependency and positive/negative contribution relations. These links clarify the relation between different qualities (e.g., security, privacy, usability, etc.) and their contribution to the overall trustworthiness goal.

As for BPMN, we use it because it is a widely-accepted business process modeling language, and it fits our needs very well. However, any other business process modeling language that supports the needed concepts can be used instead of BPMN.

In summary, we selected BPMN and i* because these two languages fit well with our conceptual model. However, our proposed framework is independent of particular languages. The framework including the conceptual model (Section 3) explains why it is beneficial to combine goal and business process modeling in general (without considering specific languages). Our method (Section 4) then shows how this combination can look when selecting BPMN and i*.

3. Framework for Systematic Analysis and Modeling of Trustworthiness Requirements

Our framework is employed to decompose high-level goals into low-level goals. We shape and structure our framework (shown in Figure 1) based on the twin peaks model [14]. The cornerstone of embedding the development of business information systems in the twin peaks model is that requirements engineers and designers co-develop a system's requirements and its architecture specification concurrently and iteratively. The same applies to our proposed approach for the analysis of trustworthiness requirements and the integration of them into business models.

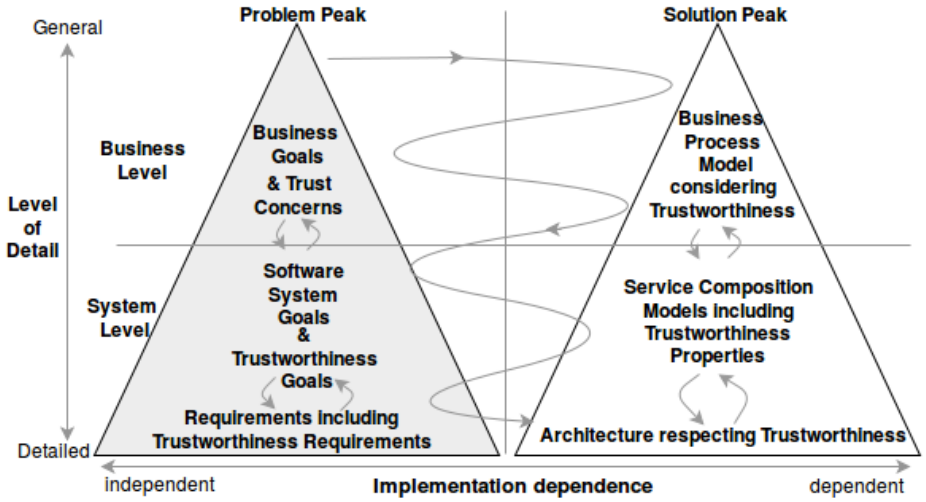


Figure 1. Overview of the proposed framework inspired by [14].

The framework captures the progression and refinement from general to detailed understanding and expression of both requirements and design. The problem peak is independent of the technical description related to the implementation of the requirements.

The solution peak incrementally introduces technical implementation details. In the problem peak, we see that business goals and trust concerns are refined to system/software goals and trustworthiness goals and finally to requirements including trustworthiness requirements. In the solution peak, the business process model that considers trustworthiness is refined to service composition models, which include trustworthiness properties. These may be further refined into an architecture respecting trustworthiness.

In order to produce the artifacts shown in our framework, we suggest a method that is described in Section 4. However, other methods may also be applied to produce them. Our method starts with the problem peak. Yet, one may also start with the solution peak.

The method for eliciting and refining trustworthiness requirements is the combination of goal models (to say what, problem peak) and business process modeling (to show how, solution peak). There should be a rationale about where a trustworthiness requirement originates from. To this end, the most appropriate level is considering trust concerns with goals of other actors starting from the business level. Our framework aligns organizational (business) requirements in an adequate way with trustworthiness requirements. The framework supports elicitation and refinement of system-level goals and trustworthiness requirements from business goals and trust concerns at the business level. Furthermore, the framework tackles the problem of high-level and low-level trustworthiness requirements' misalignment between the business/organizational level and the application and software service level. The elaborated business process model in the solution peak satisfies business goals, as well as trustworthiness goals.

Figure 2 shows how our proposed framework streamlines trustworthiness to the software development. The left side of Figure 2 shows the level of abstraction for trustworthiness and its influence on the system side on the right side (simplified service-oriented architecture layers [15,16]). The participants of a business process are presented on the context layer. In a business process, an end-user has different trust concerns than an organization. The business goal of an organization might even be in conflict with the trust concerns of an end-user. The legislator may affect both business goals and trust concerns.

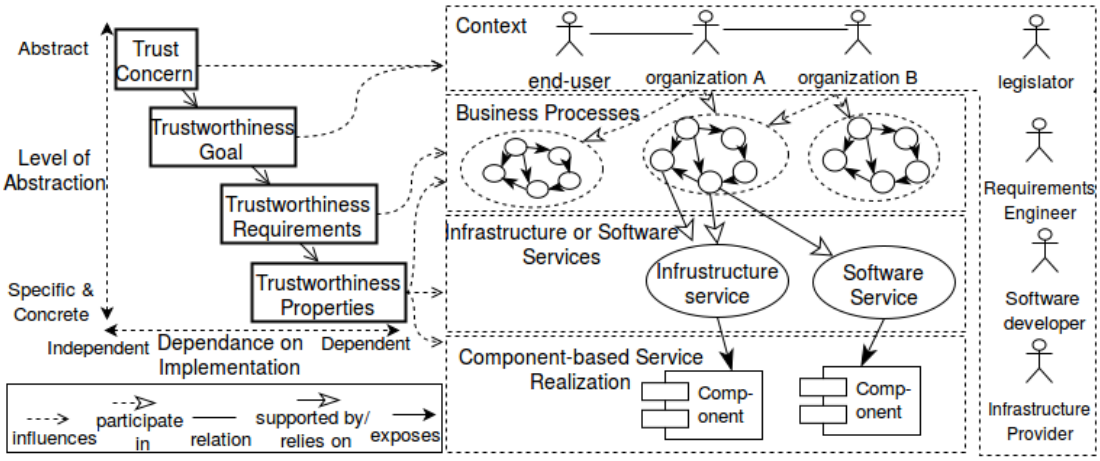


Figure 2. Placing our proposed approach for enriching business processes with trustworthiness requirements and their alignment with software development.

Conceptual Model. Since our framework suggests to refine trustworthiness requirements by combining goal models and business process models, we have created a conceptual model that supports combining existing goal modeling and business process modeling languages. The conceptual model is given as a Unified Modeling Language (UML) class diagram in Figure 3.

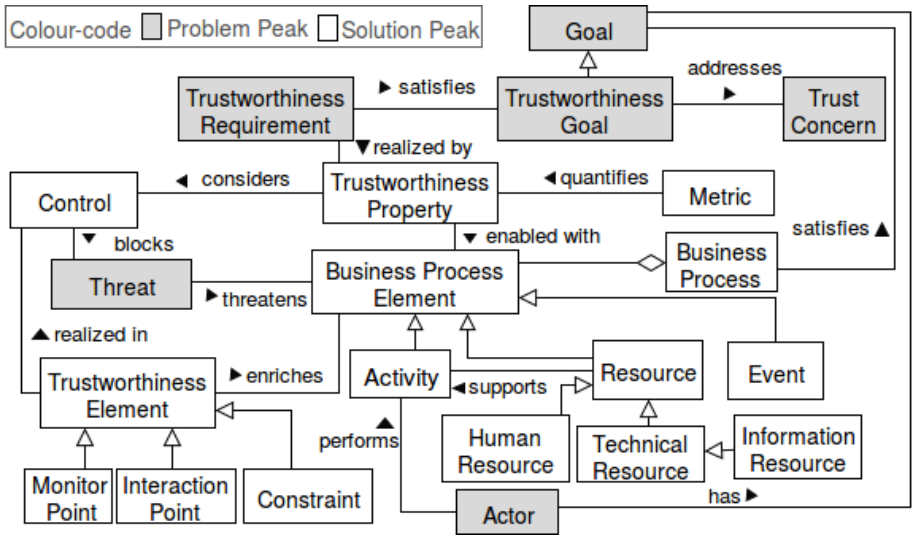


Figure 3. Conceptual model of our proposed framework and the method.

A *trustworthiness goal* is a special *goal* that addresses the trust concerns of users. A trustworthiness goal is satisfied by *trustworthiness requirements*, which can be realized by more concrete *trustworthiness properties*.

Actors have goals that can be satisfied in a *business process*. A business process consists of *business process elements*: a set of activities, events and involved resources. Here, activity, resource or event are more concrete *business process elements*. An actor performs an *activity*. An activity is supported by resources. For instance, an activity consumes data objects (information resource) as input or technical resources support performing an activity such as software services and applications. We use the term *business process element* to distinguish between the business process element and the *trustworthiness element*. A *trustworthiness element* can pertain to a type of *business process element*.

A *trustworthiness element* enriches a *business process element* by defining either a *monitor point*, an *interaction point* or a *constraint* on a *business process element*. For instance, a trustworthiness element may be a constraint on an activity, which is trustworthiness-related. Enriching an activity with a constraint enhances the refinement of the activity with respect to trustworthiness. We call an activity that is enriched with a constraint a trustworthiness-related activity. In general, a business process element that is enriched with a trustworthiness element is trustworthiness related. A notification for satisfying transparency can be defined as a constraint on an activity. A *monitor point* marks the start and the end point in a business process that shall be monitored during run-time. In a monitor point, we can not only specify which part of the process needs to be monitored during run-time, but also the desired behavior by adding the constraint. An *interaction point* marks the elements of a business process, which are in direct interaction with the end-user. These *enriched relations* enable us to derive trustworthiness requirements in the form of commitments reached among the actors for the achievement of their goals.

A *threat* is a situation or event that, if active at run-time, could undermine the trustworthiness by altering the behavior of involved resources or services in the process. A *control* aims at blocking a threat. A *metric* is used as a function to quantify trustworthiness properties. Trustworthiness elements realize the control in terms of defining elements that address the trustworthiness, e.g., an additional activity can be defined to block a threat to privacy. These additional activities could involve documenting or triggering a notification upon a delegating case of a patient to another authority or an engagement of a new service from a new third party.

In our conceptual model, we use the basis concepts used in goal modeling and show how to align them with process model components. As mentioned earlier, every goal modeling language that supports these concepts can be used. The same is valid about business process modeling languages.

4. The Method for Systematic Analysis of Trustworthiness Requirements

Here, we describe our proposed method for the refinement of trustworthiness requirements. Our method uses goal and business process modeling, iteratively. An application example in Section 5 exemplifies the steps of our method. The steps are as follows:

Step 1. Context analysis: The first step is concerned with identifying the participants and initial context information. The latter can be captured in a context model. The context information provides an overview of the process, as well.

Step 2. Set up the goal model: This step is concerned with setting up the goal model by capturing the major intentions of the involved participants/stakeholders. The goals are captured either by interviewing involved stakeholders or are based on the expertise of a requirements engineer or business engineer at the business level. We start with high-level goals and then refine them within the problem peak. The basic elements of the i* modeling notation are shown in Figure 4.

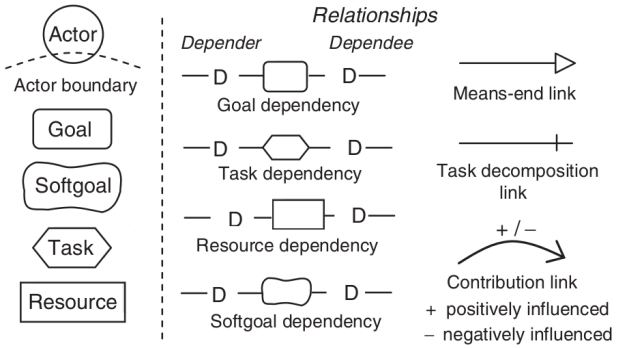


Figure 4. Notation of the i* modeling elements.

We model and document the goals in the SDM (cf. Figure 7) and SRM (cf. Figure 9) models. Please note that from the SRM in Figure 9, only the white-colored elements exist at this point of time.

Step 3. Set up the business process models: As input, the SDM and SRM models are used. We select a specific goal from the SDM. For satisfying the selected goal, we set up a business process model. As notation, we use BPMN. The basic elements of the BPMN notation for business process modeling are shown in Figure 5. To create the business process model, we use information shown in the SDM and SRM. Based on the SDM, the dependency between roles and other goals can be analyzed. SRM models give insight into the resources and activities. The business process model (cf. Figure 8) for a specific goal visualizes the control and data flow between identified tasks, used resources and involved actors.

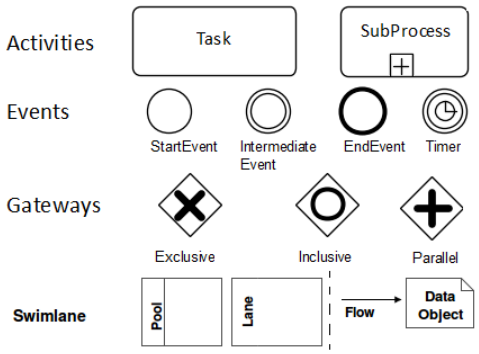


Figure 5. The basic elements of the BPMN.

Step 4. Identify trust concerns: Trust concerns of end-users and their dependencies on other participants in the business are identified. Trust concerns can be collected either by interviewing involved end-users or are based on the expertise of a requirements engineer. Prior to this step, the participants of a business and stakeholders are captured. We assume that this information about the context is provided in a context model. Trust concerns are subjective. To support this step (especially considering subjectiveness of trust), a questionnaire is provided in our previous work [17].

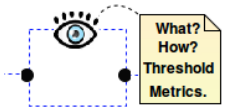
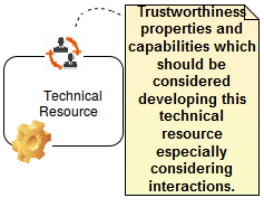
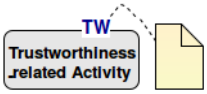
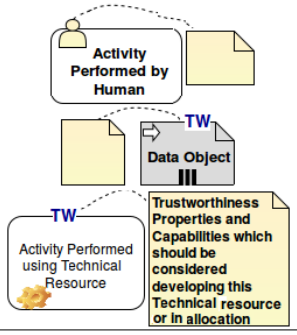
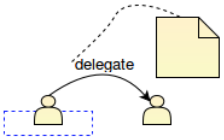
Step 5. Goal model including trustworthiness goals: Based on trust concerns, we refine the goal model with the trustworthiness goals and their relation to the other goals (negative or positive influences). The trustworthiness goals include the purpose for incorporating trustworthiness properties into the system under development. To support this step, a catalog of trustworthiness attributes that contribute to mitigate trust concerns is provided in our previous work [18]. Figure 9 shows such a goal model. Please note that the gray-colored elements were added to the goal model for addressing trustworthiness.

Step 6. Business process model including trustworthiness properties: Enhance the business process model from Step 3 by adding trustworthiness properties, which fulfill the trustworthiness goals. For supporting this step, we provide the new trustworthiness elements (cf. Figure 3). The new elements and the notation we suggest are shown in Table 1.

We propose a BPMN extension that allows the integration of trustworthiness requirements into a business process. We introduce trustworthiness elements for business process modeling, which allows modeling and documenting trustworthiness requirements, as well as placing a control or putting constraints on the resources used in the business process to address the trust the concerns of the end-users. We list our new elements (cf. Table 1) that enrich business process elements in documenting the trustworthiness requirements as follows:

- Monitor points: We introduce the monitoring points with start and end points in the process model for monitoring and the trustworthiness properties that must be considered in the defined monitored points, as well as the desired/target values for them. Furthermore, the metrics can also be provided for quantifying trustworthiness properties that will be under observation at run-time. Monitor points can be used in combination with constraints to express the desired values and metrics for measuring trustworthiness properties at run-time.
- Interaction points: These points specify the interfaces where the end-user is involved in the business process, e.g., she/he may interact with the technical resources (e.g., apps, software services) that support her/him in performing her/his tasks. In these interfaces, there are factors that could signal the trustworthiness of the system to the end-user, e.g., reliability, quality of visualization, usability, understandability of represented information, quality of service like availability or response time. For example, if an elderly person uses an app for reviewing his/her medical plan and medication, the visualization of his/her health status and medical plan influences her/his trust about the correctness of those health reports, medications or medical plans. Therefore, the trustworthiness requirements in these points need to be investigated further, and the resources involved in these points should include related trustworthiness properties that satisfy the trustworthiness requirements.
- Trustworthiness constraints: In addition to new elements like monitor and interaction points, each BPMN element can be enriched/annotated with the constraints that it should keep for satisfying trustworthiness requirements. The action with trustworthiness requirements and constraints is tagged with "TW" in the business process model, e.g., time constraints on activities or constraints on the resources that are used in performing a specific activity.

Table 1. Extended elements to model trustworthiness requirements in BPMN.

Defined Trustworthiness Element (Extension)		Definition	Symbols
Monitor Point		Inserting monitor points into the business process defines the start and end point of monitoring at run-time. Monitor points can be used in combination with constraints to express the desired values and metrics for measuring trustworthiness properties at run-time.	
Interaction Point		Interaction points are the places where the end-user interacts with the system. The interaction is normally supported by the apps or softwareservices. Qualities of these apps and software services have an impact on the trust perceptionof users. Therefore, it should be studied well how to signal their trustworthiness to the end-user. Interaction points can be further detailed in combination with constraints on technical resources (in interaction points), e.g., specifying which quality, to what extent (e.g., 99% availability).	
Constraint	Constraints on Activity	Trustworthiness requirements on a specific activity, e.g., expected duration of an activity.	
	Constraints on Resources	Trustworthiness requirements on a specific resource (either human or non-human), e.g., expertise of the involved human resource.	
	Constraints on Delegations	Trustworthiness requirements on delegation, e.g., if a delegation (e.g., activity delegation) is allowed or delegation to whom or which roles are allowed.	

The business process model from Step 3 is analyzed by identifying which business process elements are related to the identified trustworthiness goals from Step 5. We select one of the business process models for including trustworthiness requirements satisfying trustworthiness goals. This selection is based on the relation of the trustworthiness goal to the other goals. This step goes through business process elements and control flow and questions whether the element in the business process is trustworthiness related. The relation of trustworthiness goals in the goal model to the other goals from Step 5 assists this step.

Step 7. Refinement of goal model (problem peak): Refine the goals and trustworthiness goals further in order to obtain user-centered trustworthiness requirements on resources and tasks. This refinement is performed within the problem peak. However, based on the output of this step, revisions of business process models can be necessary.

Step 8. Refinement of business process model (solution peak): Detail the business processes by including trustworthiness properties on resources, activities, etc., for satisfying trustworthiness requirements. This refinement is performed within the solution peak. However, based on the output of this step, revisions of goal models can be necessary. Refinement of the business process model details business processes by including more concrete trustworthiness properties on business process elements. This step can be performed concurrently to the goal and trustworthiness goal refinements, and both models can develop iteratively.

5. Application Example

This section demonstrates our approach of eliciting and refining trustworthiness requirements and specifying trustworthiness properties on business process elements. The example stems partially from the experience that the first author gained during the OPTET project on an Ambient Assisted Living (AAL) system.

Motivating Scenario

In our scenario, Alice is an elderly person who lives alone in her apartment. She does not feel comfortable after a heart attack. She was unconscious in her home for several hours. Alice has been informed that there are some AAL services available in the marketplace. She considers using one of those services to avoid similar incidents in the future. She desires an AAL service that will suit her specific needs. We illustrate in Figure 6a general approach using supporting tools and provided apps to perform the activities. We assume that some of these software services are to be built by software developers, who will also benefit from the results of our work in developing trustworthy apps, software services, etc.

Step 1. Context analysis: We will focus on a Home Monitoring System (HMS) for incident detection and detection of abnormal situations to prevent emergency incidents. The home monitoring system allows elderly people in their homes to call for help in case of emergency situations. Furthermore, HMS analyzes the elderly person's health status for preventing incidents in the first place. The incidents are reported to an alarm call center that, in turn, reacts by, e.g., sending out ambulances or other medical caregivers and notifying the elderly person's relatives. For preventing emergency situations, the vital signs of the elderly person are diagnosed at regular intervals to reduce hospital visits and falls. Figure 6 shows an exemplary design-time system model including physical, logical and human resources/assets.

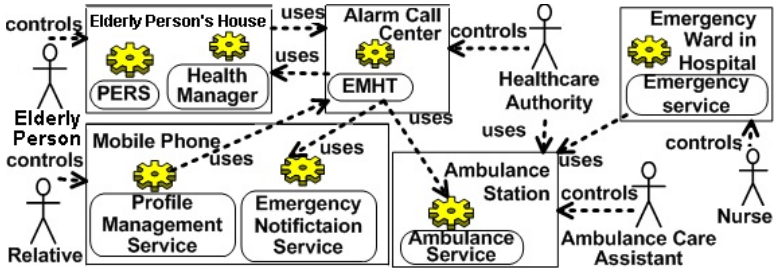


Figure 6. Part of the home monitoring system for handling healthcare cases inspired by [1].

Using this system, an elderly person uses a Personal Emergency Response System (PERS) device to call for help, which is then reported to the alarm call center that uses an Emergency Monitoring and Handling Tool (EMHT) to visualize, organize and manage emergency incidents. Furthermore, elderly persons are able to use a Health Manager (HM) app on their smart device for organizing their health status like requesting healthcare services or having an overview regarding their medication or nutrition plan. The EMHT is a software service hosted by the alarm call center that, in turn, is operated by a healthcare authority. Emergency notification and ambulance service, which run on mobile phones of relatives, or ambulance stations respectively, are called in order to require caregivers to provide help. An ambulance service is requested in case an ambulance should be sent to handle an emergency situation. The other case is that, based on the analyzed information sent to the EMHT, an abnormal situation is detected, and further diagnoses are necessary. Therefore, the elderly person will get an appointment and notifications for a tele-visit in her health manager app.

Step 2. Set up the goal model: Figure 7 captures the goals of different participants and their dependencies on each other or the realization of the goals. This is done based on the

expertise of a requirements engineer and the knowledge gained during the context analysis, for example, by means of interviews. Here, we only focus on the elderly person and the alarm call center. The ambulance station is also involved, because for handling the emergency cases, the alarm call center is dependent on the ambulance as a resource.

Additional to the SDM presented in Figure 7, we have further SRM models that provide more detail on tasks, resources and soft goals within the actor boundaries. As an example, one can consider the SRM model in Figure 9. In this step (i.e., at this point of time), we have only the white-colored elements of that SRM.

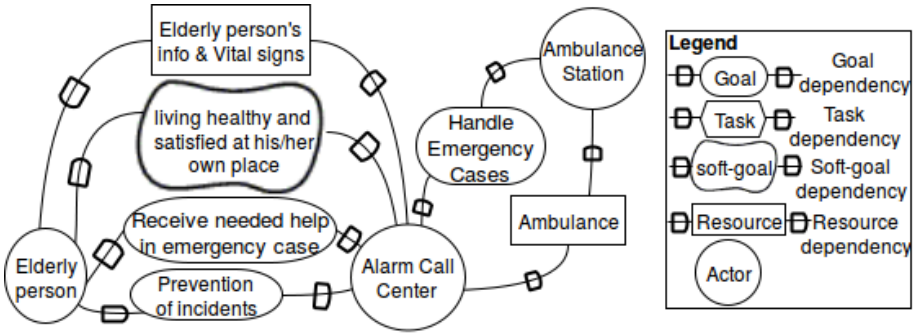


Figure 7. Simplified Strategic Dependency Model (SDM) with the dependencies between identified participants.

Step 3. Set up the business process model: Figure 8 illustrates and exemplifies the typical steps that, for example, caregivers in an alarm center have to take once they notice that the health record of an elderly person deviates from the normal situation and further examination is needed. This business process model targets the satisfaction of the goals reducing hospital visits and prevention of incidents (cf. Figure 7).

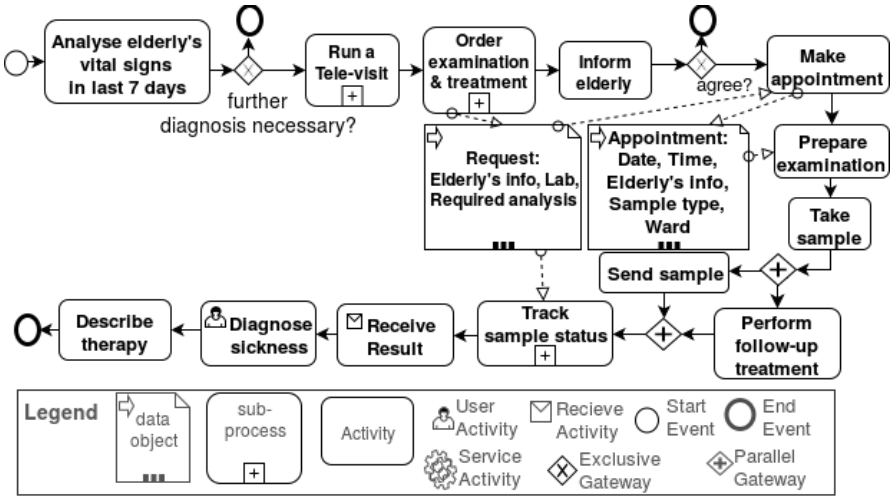


Figure 8. Exemplary business process model for preventing emergency cases and reducing hospital visits.

The process starts by analyzing the elderly person's vital signs in the last seven days. These data are examined by a physician, who decides whether the elderly person is healthy or additional examination needs to be undertaken. In the former case, the physician fills out the examination report. In the latter case, a tele-visit is performed by this physician in which the physician informs the elderly person about the examination and necessary treatment. An examination order is placed by

the physician. The physician sends out a request. This request includes information about the elderly person, the required examination and possible labs. Furthermore, an appropriate appointment should be arranged.

The process continues for taking a sample and validating this. Eventually, the physician from the alarm call center should get the result in order to make the diagnosis and prescribe the medication.

Step 4. Identify trust concerns: Alice is concerned about whether she will really receive the emergency help if a similar situation happens again (heart attack experience). Alice is informed that by using the HMS, she can have regular diagnoses, which can prevent frequent hospital visits. However, Alice is concerned whether she will be able to use the service in a proper way. She is also concerned about who can get access to the data about her diseases or life habits. She indicates that she would only prefer her regular nurse and doctor to be able to see her history and health status.

Step 5. Goal model including trustworthiness goals: Based on the trust concerns, a requirement engineer adds trustworthiness goals to the goal model. The existing goal-based refinement techniques are applied to refine these trustworthiness goals into trustworthiness requirements. Considering the healthcare domain, reliability, availability, usability, raising awareness and privacy (providing guidance and users' data protection) are crucial issues related to trustworthiness [19]. For instance, electronic medical transactions require the transmission of personal and medical information over insecure channels, e.g., the Internet. Patients' profiles document the medical behavior of patients or even include sensitive information, e.g., their medical history.

Considering the trustworthiness of a healthcare application, one can consider a vector of multiple trustworthiness goals. They either address the fulfillment of the mission, e.g., reliability, availability when the patient needs help, correctness of prescribed therapy, or address it from a privacy perspective. The gray-colored soft goals in Figure 9 are the trustworthiness goals added to the goal model in this step.

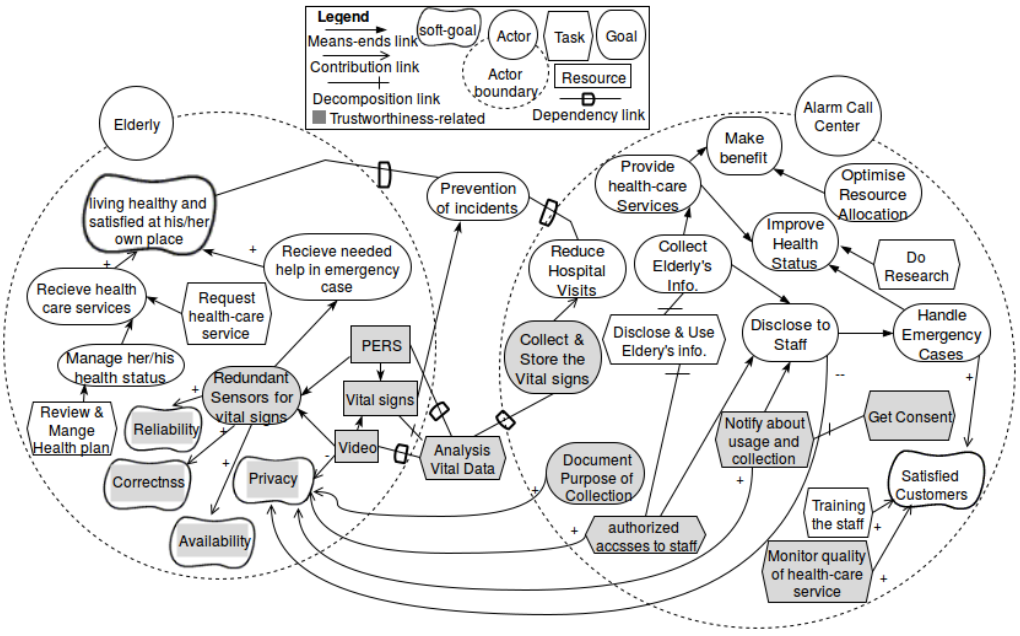


Figure 9. Simplified Strategic Rationale Model (SRM) including trustworthiness goals considering trust concerns.

Step 6. Business process including trustworthiness properties: Figure 10 illustrates the enriched business process model with the trustworthiness requirements satisfying reliability and privacy (cf. Figure 9).

In particular, we exemplify the typical steps that a human resource (e.g., a caregiver in the alarm call center) has to take or properties that a non-human resource needs to have in order to contribute to trustworthiness. We start with the activity to analyze the history of the vital signs of the elderly person in the last seven days. This activity may detect a risk in her health status.

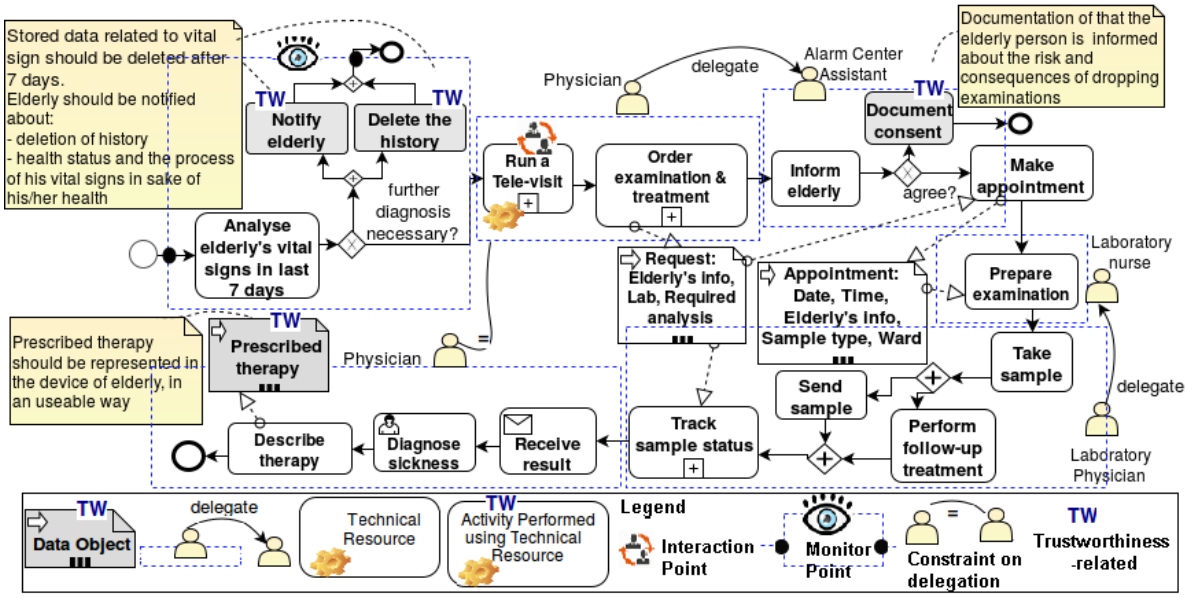


Figure 10. Exemplary business process model enriched with trustworthiness requirements.

For addressing the trust concerns of the elderly person related to her confidence that she is not left alone and will get the needed healthcare when necessary, and related to her privacy concerns, the following trustworthiness requirements are specified: The elderly person should receive a regular notification that informs her about the diagnoses that are performed on her vital signs. In Figure 10, it is added as a trustworthiness-related activity, namely "notify elderly". This activity contributes to make her confident that she is not left alone without care. Due to the privacy concerns, the history should be deleted, if no further diagnosis is necessary. The "delete the history" activity is also a trustworthiness-related activity added to the initial business process. This part of the business process is annotated as relevant for monitoring at run-time.

If a risk to the elderly person's health status is detected, a "run a tele-visit" is offered. This activity is an interaction point supported by the HM app as a technical resource (cf. Figure 10). The trustworthiness properties for this interaction point are usability, response time, etc. In case of the necessity for further examination, the elderly person should be contacted by her physician or responsible care assistant (delegation of physician to the assistants). Furthermore, based on history, the same physician should be assigned to the activities when the elderly person is in contact with the alarm call center staff (addressing the trust concern). After processing her history data and if everything is alright, her last seven days of vital signs should be deleted. She should be informed that the processing has been performed and her health status is fine. She should be informed about the deletion of her history, as well.

In Step 7 and Step 8, further iterative refinements of trustworthiness goals and business processes are performed. Gray-colored elements (additional to the elicited trustworthiness goals) in Figure 9 are the results of the refinement of the goal model. For instance, in order to satisfy reliability and

availability, the redundant sensors for sending vital signs are considered for providing the vital signs of the elderly person to the alarm call center. The task, notify about usage and collection, is added to positively influence privacy. These refinements are further elaborated in the business process models.

Figure 11 shows further the refinement of the trustworthiness requirements related to the notify elderly activity, which is related to the notify about usage and collection from the goal model (cf. Figure 9).

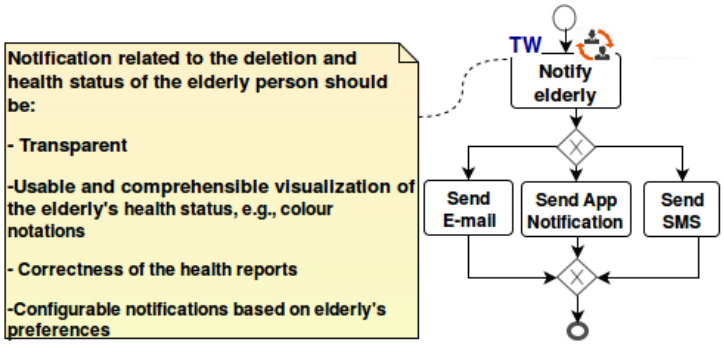


Figure 11. Exemplary further refinement on the business process model (within the solution peak).

Table 2 shows the trust concerns, corresponding requirements and activities. The column "affected resources" exemplifies possible software design decisions on resources.

Table 2. Examples of captured trustworthiness requirements and properties in the business process and directions on the design decisions.

Trust Concerns	Trustworthiness Requirements	Activities	Affected Resources
Privacy	Transparency, intervenability	Storage, deletion within 7 days, update	Private inventory system from the alarm call center, external cloud storage
Awareness	Usability, transparency, reliability, availability	Notifications, place appointments	App on the elderly person's smart device (HM)
Safety, reliability	Reliability, availability	Raise alarm	Redundant sensors in addition to PERS
Privacy	Correctness, usability, availability	Make appointment, prescribe examination	Elderly person's details

6. Benefits of Our Method

In this section, we present some of the benefits of applying our method. The elicitation and specification of trustworthiness requirements through a user-centered requirements analysis and modeling method using a combination of goal and business process modeling is an innovative proposal.

- Elicitation of trustworthiness requirements with a direct link to the originating trust concern and using them to make informed design decisions is a key success factor for developing a trustworthy system.
- Bridging the two peaks (problem and solution) helps to elaborate the synergies between requirements and design artifacts. The benefits of interleaving the tasks of eliciting and refining requirements with the tasks of designing a software solution has long been recognized [20]. Trustworthiness requirements together with other requirements are evolutionarily developed and will not be ignored. In a development process that incorporates just an up-front requirements engineering method, there is the risk of ignoring trust concerns. Developers might deliver solutions that fail to treat trust concerns of end-users. It is therefore important to proactively elicit

trustworthiness requirements from different participating actors during early phases and then to consider design solutions that balance and satisfy those concerns.

- Combining goal models and business process models has the benefit that we avoid gold plating. Gold plating [21] is known as the implementation of irrelevant requirements, i.e., requirements that were never requested by stakeholders and are thus not backward traceable to any goals of stakeholders. Yet, our trustworthiness requirements can always be traced back to trust concerns of stakeholders. Therefore, there is always a justification for the trustworthiness requirements that are realized. This becomes even more critical, when the users pay it with the cost of their privacy or sometimes harming trust. For instance, in the home monitoring system, when the elderly person does not desire any assistance outside her/his home, the provided app (e.g., health manager) should not be permitted on location information.
- The explicit consideration of conflicts on the goal level is another benefit of our approach. Sometimes some of the end-users' trust concerns are in conflict with the goals of other involved actors. For instance, the privacy goal of an elderly person in terms of transparency might be in conflict with the business goals of the hospital or insurance company. To resolve this conflict, the extent of transparency can be negotiated between the parties. In our approach, such kinds of contradictions are dealt with before the developers go into technical realizations of the goals.
- Dependencies between actors together with positive and negative links in i* [6] describe the trustworthiness-related dependencies. However, dependencies and contribution links do not capture business relationships. These relationships are described in the business process models in our approach. One may understand the business process models also as an architectural description of the application.
- The resulting business process models with specified trustworthiness requirements can be used as a basis for designing and developing trustworthy software systems, applications and even for the evaluation of the trustworthiness properties [1] (e.g., privacy, reliability, confidentiality or integrity) on an abstract level.
- Business process modeling offers an appropriate abstraction level to describe trustworthiness requirements and to later evaluate trustworthiness-related risks.

7. Related Work

Systematic consideration of trustworthiness requirements is one of the key challenges that trustworthy systems must meet. However, most of the existing approaches restrict trustworthiness to security and/or privacy. Yet, in our opinion, trustworthiness is a vector of different other qualities and properties. Nevertheless, we present and discuss the most important existing approaches in this area. First, we discuss goal modeling approaches. Second, we discuss business process modeling approaches. Then, we discuss the approaches that use combinations of goal and business process modeling.

Horkoff et al. [22] provide a systematic literature review of goal-oriented requirements engineering. They give a high-level overview of the goal-modeling field. According to their systematic literature review, i* and KAOS are the most dominant goal modeling languages used for requirements engineering.

Both KAOS and i* have been extended by researchers to consider security and privacy requirements. An extension to KAOS for considering security requirements is presented by van Lamsweerde [23]. van Lamsweerde defines the concept of anti-goals to elicit the security goals.

An extension of the i* approach for considering security and privacy requirements is developed by Liu et al. [24]. They support the modeling of the social context of software systems and the identification of malicious intents toward the system under consideration. This work suggests a methodological framework for dealing with security and privacy requirements. Soft goals are used to model the corresponding notions of security and privacy within an agent-oriented modeling.

Secure Tropos [25], which extends the Tropos methodology [26], is an agent-oriented software development methodology tailored to describe functional, as well as security and trust requirements. However, in this approach, security is considered as the only dimension of trust. A Computer-Aided

Software Engineering (CASE) tool for design and verification of functional, security and trust requirements is presented by Giorgini et al. [4]. This CASE tool supports the Secure Tropos methodology.

Mellado et al. [27] carry out a systematic literature review on security requirements engineering. They summarize the approaches considering security in all stages of information system development, especially during the requirements engineering phase.

Indeed, there is plenty of work regarding requirements refinement processes considering trust. Therein, first, systems are specified consisting of functionalities with no security features. Then, the next refinement includes encryption, access control and authentication. However, there is no clear reasoning why encryption, access control and authentication are necessary and sufficient with respect to trust.

An extension of problem frames [28] has been developed for the elaboration and analysis of security requirements. Trust assumptions are made explicit in problem diagrams [3]. Yet, this approach focuses more on security requirements and detailing the context information based on trust assumptions.

De la Vara and Sánchez use BPMN to specify requirements in general. The requirements are specified by means of the description of the business processes to be realized by the system under development [29].

The study of related work reveals also some gaps in business process management with respect to trustworthiness.

Short et al. [30] provide an approach for dealing with the inclusion of internal and/or external services in a business process that contains data-handling policies. Wang et al. [31] suggest a method to govern adaptive distributed business processes at run time with an aspect-oriented programming approach. Policies can be specified for run-time governance, such as safety constraints and how the process should react if they are violated. Several works have been done to overcome the problem of considering qualities in resource assignment. Some meta-models like [32,33] and an expressive resource assignment language [5] have been developed. Among those, Resource Assignment Language graPH (RALPH) [5] provides a graphical representation of the resource selection conditions and assignments. RALPH has formal semantics, which makes it appropriate for automated resource analysis in business process models.

There are a number of different extensions to BPMN. Stroppi et al. [34] present an extension for providing flexibility for resource structuring, allowing the definition of a broad range of organizational structures.

Stepien et al. [35] present user interfaces that users can use to define conditions as policies themselves, e.g., defining privacy policies. The resource patterns provided by Russell et al. [36] are used to support expressing criteria in resource allocation.

Business activities is a role-based access control extension of UML activity diagrams [37] to define the separation of duties and binding of duties between the activities of a process. Wolter et al. [38] developed a model-driven business process security requirement specification, which introduces security annotations into business process definition models for expressing security requirements for tasks. However, the current state of the art in this field neglects considering trustworthiness as a criterion for the resources and business process management.

There are BPMN extensions for the inclusion of different security requirements, e.g., non-repudiation, attack harm detection, integrity and access control [39,40]. There are also proposed languages for the formulation of security constraints embedded in BPMN [41].

Salnitri et al. [42] develop the Secure BPMN (SecBPMN) modeling language that extends BPMN with security annotations. SecBPMN supports establishing compliance between security-annotated business processes and security policies.

In all of these approaches, only security requirements are incorporated into a BPMN process from the perspective of a business process analyst.

Transforming goal models into process models provides the rationale for the design choices. Horkoff et al. [43] conducted a systematic literature review. They created a road-map of publications that transform goal models to/from other software artifacts. According to their study, goal models are transformed into business process models in order to perform business analysis. However, our work is not transforming goal models to business process models in an automatic way. Rather, we use the combination of goal and business process models for eliciting, refining and modeling trustworthiness requirements.

Bleistein et al. [44] use the Goal-oriented Requirement Language (GRL) to link requirements for strategic-level e-business systems to business strategies and to document patterns of best practices. They explore goal modeling for providing traceability and alignment between strategic levels, tactical and operational ones.

The linkage between goal and business process models is also applied in the context of security and privacy. For instance, the work by Salnitri et al. [45] incorporates such an approach in the context of security. Kalloniatis et al. [46] developed the Privacy Safeguard (PriS) method. The PriS method applies the linkage between goal and business process models for privacy requirements. PriS is a security requirements engineering method that incorporates privacy requirements into the system development process. PriS considers privacy requirements as organizational goals and uses privacy process patterns in order to describe the effect of privacy requirements on business processes. PriS captures the organizational goal in the form of goal trees, where the leaves of the tree refer to the processes that should satisfy the goal. Processes are defined in the form of activity diagrams. Argyropoulos et al. [47] also apply the transformation of goals to service level process models. They provide a semi-automatic approach for the derivation of cloud service requirements focusing on security and privacy.

In our work, we consider a broad range of trustworthiness properties rather than just security or privacy. Furthermore, there is a rationale about from where trustworthiness requirements originate. Our approach tackles the problem of misalignments of trustworthiness requirements between the business and the system level.

8. Evaluation Plan

The proposed method was used for analyzing trustworthiness requirements in an ambient assisted living system.

A comparative evaluation of the proposed method and the state of the art requirements engineering methods that consider trustworthiness is planned. We plan an evaluation with some groups of students with a real-life example from a smart grid system.

To use energy in an optimal way, smart grids make it possible to couple the generation, distribution, storage and consumption of energy. Smart grids use information and communication technology, which allows financial, informational and electrical transactions. As information sources, we will consider diverse documents, such as "Application Case Study: Smart Grid" and "Smart Grid Concrete Scenario" provided by the industrial partners of the NESSoS project (http://www.nessos-project.eu/).

The study will be a qualitative research experiment. One group of students will prepare a requirements specification using the Secure Tropos methodology as a goal modeling approach dealing with trust. Another group will use business process modeling as an approach for the specification of requirements. Please note that here also one of the extensions that targets trustworthiness will be applied. Another students' group is free to select an arbitrary approach like the extension of problem frames for dealing with trust. One of the students' groups will use our presented approach. The result of their practice will be qualitatively evaluated by an expert in requirement engineering.

The effort of preparing the diagrams will also be measured. The effort per item and the total effort regarding the number of items can be measured by "spent hours". These measures can also be used in our comparative study. This way, we can provide the information about the effort of documenting specification and their quality in documenting trustworthiness requirements along with the other

requirements. The number of documented trustworthiness requirements can also be considered as relevant.

Validating the actual applicability and acceptance of our method in industry using an experiment with the industrial partners of the HORIZON 2020 project "RestAssured" (http://cordis.europa.eu/project/rcn/206344_en.html) is planned. The experiment will evaluate the method with regard to its usefulness, ease-of-use, learnability, compatibility and value for practitioners. The results will be published in a technical report.

9. Conclusions

Managing business processes respecting trustworthiness requirements remains an ongoing challenge in service-oriented computing and cloud computing research.

This paper discussed trust issues in the context of business process management using BPMN and i*. Goal modeling provides an explicit linkage between requirements at different levels of abstraction as determined and refined by the i* diagrams. The integration with business process models offers a means of helping to ensure that requirements are in harmony with and provide support for business goals. An integration of subjective trust concerns into goal models and thereafter into the process models is provided. Our framework supports the analysis of the business processes from activity, resource and data object perspectives with respect to trustworthiness. To the best of our knowledge, we propose a novel contribution on user-centered identification of trust concerns and elicitation of trustworthiness requirements and thereafter integrating trustworthiness properties into business process design. Furthermore, our contribution includes a preparation for verification that satisfies trustworthiness constraints over resource allocation and activities execution.

Trustworthiness requirements are usually defined first on a technical level, rather than on a business process level. However, at the business process level, we are able to provide a comprehensive view on the participants, the assets/resources and their relationships regarding the satisfaction of business goals, as well as trustworthiness goals. Integrating trustworthiness-related information into business processes will support designers and developers in making their design decisions. Trustworthiness requirements on the business process level can be translated into concrete trustworthy configurations for service-based systems. Therefore, our proposed approach can be applied on different abstraction levels. Our proposed method identifies the resources and activities that are trustworthiness-related. Then, we specify the trustworthiness requirements on those resources and activities in business processes with regard to trustworthiness goals from goal models. Furthermore, our framework supports the business process life-cycle with respect to trustworthiness.

This paper is limited to BPMN and i* as the languages used. However, our approach is not limited to using i* as the goal modeling technique. For goal modeling, i* and KAOS are considered, which are used in our analysis due to their comprehensiveness.

This is a work-in-progress paper. The main ideas and findings will be further investigated and evaluated based on the presented example in Section 5. This will lead to the establishment of patterns and metrics for trustworthiness. To reduce the process designer's effort, we plan developing a set of patterns for easing trustworthiness requirement specifications. Our future research will focus on three important issues: (1) the proposed method needs a full integration to a business process modeling or management application; (2) inter-model dependencies between different models, tracing and justification of whether a trust concern is addressed should also be provided with a tool support; (3) investigate existing risk assessment methodologies on the business process level and show how they can support business process design and definition in building trustworthiness into processes in the whole life-cycle of business process management.

We will perform our evaluation plan presented in Section 8. Based on the result of the planned experiment, we will improve our understanding and encourage the utilization of our framework and method.

Acknowledgments: The research leading to these results has received funding from the European Community's Horizon 2020 Framework Programme under Grant Agreement Number 731678. We are also grateful to the reviewers of the paper for their comments. We wish to thank our colleague Nelufar Ulfat-Bunyadi for the proof-reading and providing expertise in the discussions of the paper. We thank all of our colleagues in the software engineering group for their feedback that helped us in improving the paper, especially Rene Meis.

Author Contributions: Nazila Gol Mohammadi is the main co-author of the paper, and Maritta Heisel provided substantial feedback that improved the paper.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

Tr	Trust
TW	TrustWorthines
BPMN	Business Process Model and Notation
SecBPMN	Secure Business Process Model and Notation
KAOS	Knowledge Acquisition in autOmated Specification
UML	Unified Modeling Language
AAL	Ambient Assisted Living
HMS	Home Monitoring System
PERS	Personal Emergency Response System
EMHT	Emergency Monitoring and Handling Tool
HM	Health Manager
CASE	Computer-Aided Software Engineering
RALPH	Resource Assignment Language graPH
GRL	Goal-oriented Requirement Language
PriS	Privacy Safeguard

References

1. Gol Mohammadi, N.; Bandyszak, T.; Kalogiros, C.; Kanakakis, M.; Weyer, T. A Framework for Evaluating the End-to-End Trustworthiness. In Proceedings of the 14th IEEE International Conference on Trust, Security and Privacy in Computing and Communications (IEEE TrustCom), Helsinki, Finland, 20–22 August 2015; pp. 638–645.
2. Gol Mohammadi, N.; Bandyszak, T.; Paulus, S.; Meland, P.H.; Weyer, T.; Pohl, K. Extending Software Development Methodologies to Support Trustworthiness-by-Design. In Proceedings of the CAiSE Forum, Stockholm, Sweden, 8–12 June 2015; pp. 213–220.
3. Haley, C.B.; Laney, R.C.; Moffett, J.D.; Nuseibeh, B. The Effect of Trust Assumptions on the Elaboration of Security Requirements. In Proceedings of the 12th IEEE International Requirements Engineering Conference, Kyoto, Japan, 6–10 September 2004; pp. 102–111.
4. Giorgini, P.; Massacci, F.; Mylopoulos, J.; Zannone, N. Requirements Engineering for Trust Management: Model, Methodology, and Reasoning. *Int. J. Inf. Secur.* **2006**, *5*, 257–274.
5. Cabanillas, C.; Knuplesch, D.; Resinas, M.; Reichert, M.; Mendling, J.; Ruiz-Cortés, A. RALph: A Graphical Notation for Resource Assignments in Business Processes. In *Advanced Information Systems Engineering, CAiSE 2015*; Springer: Berlin/Heidelberg, Germany, 2015; pp. 53–68.
6. Yu, E.S.K. Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering. In Proceedings of the 3rd IEEE International Symposium on Requirements Engineering, Annapolis, MD, USA, 5–8 January 1997; pp. 226–235.
7. Object Management Group (OMG). Business Process Model and Notation (BPMN) Version 2.0; Technical Report; Available online: http://www.omg.org/spec/BPMN/2.0/ (accessed on 17 April 2017).
8. Gol Mohammadi, N.; Heisel, M. A Framework for Systematic Analysis and Modeling of Trustworthiness Requirements Using i* and BPMN. In Proceedings of the International Conference on Trust and Privacy in Digital Business (TrustBUS 2016), Porto, Portugal, 5–8 September 2016; pp. 3–18.
9. Sztompka, P. *Trust: A Sociological Theory*; Cambridge University Press: Cambridge, UK, 2000.
10. Mei, H.; Huang, G.; Xie, T. Internetware: A Software Paradigm for Internet Computing. *Computer* **2012**, *45*, 26–31.

11. Stroppi, L.J.R.; Chiotti, O.; Villarreal, P.D. Extending BPMN 2.0: Method and Tool Support. In Proceedings of the 3rd International Workshop Proceedings of Business Process Model and Notation (BPMN); Springer: Berlin/Heidelberg, Germany, 2011; pp. 59–73.
12. Van Lamsweerde, A.; Letier, E. Handling Obstacles in Goal-Oriented Requirements Engineering. In Proceedings of the 3rd IEEE International Symposium on Requirements Engineering, Annapolis, MD, USA, 5–8 January 1997; pp. 226–235.
13. Letier, E.; van Lamsweerde, A. Agent-based tactics for goal-oriented requirements elaboration. *IEEE Trans. Softw. Eng.* **2000**, *26*, 978–1005.
14. Nuseibeh, B. Weaving together Requirements and Architectures. *Computer* **2001**, *34*, 115–119.
15. Papazoglou, M.P. Service-Oriented Computing: Concepts, Characteristics and Directions. In Proceedings of the Fourth International Conference on Web Information Systems Engineering, (WISE 2003), Rome, Italy, 10–12 December 2003; pp. 3–12.
16. Papazoglou, M.P.; Traverso, P.; Dustdar, S.; Leymann, F. Service-Oriented Computing: State of the Art and Research Challenges. *Computer* **2007**, *40*, 38–45.
17. Gol Mohammadi, N.; Heisel, M. Patterns for Identification of Trust Concerns and Specification of Trustworthiness Requirements. In Proceedings of the 21st European Conference on Pattern Languages of Programs (EuroPlop '16); ACM: New York, NY, USA, 2016.
18. Gol Mohammadi, N.; Paulus, S.; Bishr, M.; Metzger, A.; Könnecke, H.; Hartenstein, S.; Weyer, T.; Pohl, K. Trustworthiness Attributes and Metrics for Engineering Trusted Internet-Based Software Systems. In *Cloud Computing and Services Science—3rd International Conference, CLOSER*; Revised Selected Papers; Springer: Berlin/Heidelberg, Germany, 2013; pp. 19–35.
19. Avancha, S.; Baxi, A.; Kotz, D. Privacy in Mobile Technology for Personal Healthcare. *ACM Comput. Surv. (CSUR)* **2012**, *45*, 1–54.
20. Chung, L.; do Prado Leite, J. On Non-Functional Requirements in Software Engineering. In *Conceptual Modeling: Foundations and Applications*; Springer: Berlin/Heidelberg, Germany, 2009; pp. 363–379.
21. Pohl, K. *Requirements Engineering: Fundamentals, Principles, and Techniques*; Springer: Berlin/Heidelberg, Germany, 2010.
22. Horkoff, J.; Başak Aydemir, F.; Cardoso, E. Goal-Oriented Requirements Engineering: A Systematic Literature Map. In Proceedings of the 2016 IEEE 24th International Requirements Engineering Conference (RE), Beijing, China, 12–16 September 2016; pp. 106–115.
23. Van Lamsweerde, A. Elaborating Security Requirements by Construction of Intentional Anti-Models. In Proceeding of the 26th International Conference on Software Engineering (ICSE'04), Edinburgh, UK, 23–28 May 2004; pp. 148–157.
24. Liu, L.; Yu, E.; Mylopoulos, J. Security and Privacy Requirements Analysis within a Social Setting. In Proceedings of the 11th IEEE International Conference on Requirements Engineering (RE'03), Monterey, CA, USA, 8–12 September 2003; pp. 151–161.
25. Giorgini, P.; Massacci, F.; Mylopoulous, J.; Zannone, N. Requirements Engineering meets Trust Management: Model, Methodology, and Reasoning. In *Proceedings of iTrust'04, LNCS 2995*; Springer: Berlin/Heidelberg, Germany, 2004; pp. 176–190.
26. Bresciani, P.; Giorgini, P.; Giunchiglia, F.; Mylopoulos, J.; Perini, A. TROPOS: An Agent- Oriented Software Development Methodology. *JAAMAS*, **2004**, *8*, 203–236.
27. Mellado, D.; Sánchez, L.E.; Fernández-Medina, E. A Systematic Review of Security Requirements Engineering. *Comput. Stand. Interfaces* **2010**, *32*, 153–165.
28. Jackson, M. *Problem Frames: Analyzing and Structuring Software Development Problems*; Addison-Wesley: Boston, MA, USA, 2001.
29. De la Vara, J.L.; Sánchez, J. Improving Requirements Analysis through Business Process Modelling: A Participative Approach. In *Business Information Systems*; Springer: Berlin/Heidelberg, Germany, 2008; pp. 165–176.
30. Short, S.; Kaluvuri, S.P. A Data-Centric Approach for Privacy-Aware Business Process Enablement. In Proceedings of the 3rd International IFIP Working Conference Enterprise Interoperability (IWEI), Stockholm, Sweden, 23–24 March 2011; pp. 191–203.

31. Wang, M.; Bandara, K.; Pahl, C. Process as a Service Distributed Multi-tenant Policy-Based Process Runtime Governance. In Proceedings of the IEEE International Conference on Services Computing (SCC), Miami, FL, USA, 5–10 July 2010; pp. 578–585.
32. Koschmider, A.; Yingbo, L.; Schuster, T. Role Assignment in Business Process Models. In *Business Process Management Workshops*; Springer: Berlin/Heidelberg, Germany, 2012; Volume 99, pp. 37–49.
33. Van der Aalst, W.M.P.; Kumar, A. A Reference Model for Team-enabled Workflow Management Systems. *Data Knowl. Eng.* **2001**, *38*, 335–363.
34. Stroppi, L.J.R.; Chiotti, O.; Villarreal, P.D. A BPMN 2.0 Extension to Define the Resource Perspective of Business Process Models. In Proceedings of the XIV Congreso Iberoamericano en Software Engineering (CIbSE), Rio de Janeiro, Brasil, 27–29 April 2011.
35. Stepien, B.; Felty, A.; Matwin, S. A Non-technical User-Oriented Display Notation for XACML Conditions. In *E-Technologies: Innovation in an Open World*; Lecture Notes in Business Information Processing; Springer: Berlin/Heidelberg, Germany, 2009; Volume 26, pp. 53–64.
36. Russell, N.; van der Aalst, W.; ter Hofstede, A.; Edmond, D. Workflow Resource Patterns: Identification, Representation and Tool Support. In *Advanced Information Systems Engineering*; Springer: Berlin/Heidelberg, Germany, 2005; pp. 216–232.
37. Strembeck, M.; Mendling, J. Modeling Process-related RBAC Models with Extended UML Activity Models. *Inf. Softw. Technol.* **2011**, *53*, 456–483.
38. Wolter, C.; Menzel, M.; Schaad, A.; Miseldine, P.; Meinel, C. Model-driven Business Process Security Requirement Specification. *J. Syst. Archit. Spec. Issue Secure SOA* **2009**, *55*, 211–223.
39. Rodríguez, A.; Fernández-Medina, E.; Piattini, M. A BPMN Extension for the Modeling of Security Requirements in Business Processes. *IEICE Trans. Inf. Syst.* **2007**, *E90-D*, 745–752.
40. Sang, K.S.; Zhou, B. BPMN Security Extensions for Healthcare Process. In Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, (CIT/IUCC/DASC/PICOM) , Liverpool, UK, 26–28 October 2015; pp. 2340–2345.
41. Maines, C.L.; Llewellyn-Jones, D.; Tang, S.; Zhou, B. A Cyber Security Ontology for BPMN-Security Extensions. In Proceedings of the IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, 26–28 October 2015; pp. 1756–1763.
42. Salnitri, M.; Dalpiaz, F.; Giorgini, P. Designing Secure Business Processes with secBPMN. *Softw. Syst. Model.* **2017**, *16*, 1–21.
43. Horkoff, J.; Li, T.; Li, F. Taking Goal Models Downstream: A Systematic Roadmap. In Proceedings of the IEEE 8th International Conference on Research Challenges in Information Science (RCIS), Marrakech, Morocco, 28–30 May 2014; pp. 1–12.
44. Bleistein, S.J.; Aurum, A.; Cox, K.; Ray, P.K. Linking Requirements Goal Modeling Techniques to Strategic e-Business Patterns and Best Practice. In Proceedings of the I8th Australian Workshop on Requirements Engineering (AWRE'03), Sydney, Australia, 4–5 December 2003; pp. 13–22.
45. Salnitri, M.; Paja, E.; Giorgini, P. From Socio-Technical Requirements to Technical Security Design: An STS-Based Framework. Ph.D. Thesis, University of Trento, Trento, Italy, 2015.
46. Kalloniatis, C.; Kavakli, E.; Gritzalis, S. Addressing Privacy Requirements in System Design: The PriS Method. *Requir. Eng.* **2008**, *13*, 241–255.
47. Argyropoulos, N.; Shei, S.; Kalloniatis, C.; Mouratidis, H. A Semi-Automatic Approach for Eliciting Cloud Security and Privacy Requirements. In Proceedings of the 50th Hawaii International Conference on System Sciences (HICSS), Waikoloa Village, HI, USA, 4–7 January 2017.