Analyzing Business Continuity through a Multi-Layers Model

Yudistira Asnar and Paolo Giorgini

Department of Information and Communication Technology
University of Trento, Italy
{yudis.asnar,paolo.giorgini}@disi.unitn.it

Abstract. Business Continuity Management (BCM) is a process to manage risks, emergencies, and recovery plans of an organization during a crisis. It results in a document called Business Continuity Plans (BCP) that specifies the methodology and procedures required to backup and recover the functional unit of a disrupted business. Traditionally, the BCP assessment is based only on the continuity of IS infrastructures and does not consider possible relations with the business objectives and business processes. This traditional approach assumes that the risk of business continuity is resulted from the disruption of the IS infrastructures. However, we believe there are situations where the risk emerges even the infrastructures up and running. Moreover, the lack of modeling framework and the aided-tool make the process even harder.
In this paper, we propose a framework to support modeling and analysis of BCP from the organization perspective, where risks and treatments are modeled and analyzed along strategic objectives and their realizations. An automated reasoner based on cost-benefit analysis techniques is proposed to elicit and then adopt the most cost-efficient plan. The approach is developed using the Tropos Goal-Risk Framework and the Time Dependency and Recovery Model as underlain frameworks. A Loan Originating Process case study is used as a running example to illustrate the proposal.

1 Introduction

Information Systems (IS) are currently evolving in so called socio-technical systems, where human and organization factors along technical aspects assume a more and more critical role in the correct operation of the system. A socio-technical system is represented as a complex network of interrelationships between human and technical systems that includes hardware, software, users, stakeholders, data, and regulations [1]. As reported in [2], economic and social factors results being crucial in such systems and introduce challenges that lay beyond the mere technical aspects.

In sectors such as e-Banking, e-Commerce, etc., where the business strongly depends on the availability of IS's services, an organization should be able to ensure the continuity of its business objectives accordingly to the evolution of regulations (e.g., Basel II [3] or Sarbanes-Oxley Act [4]) as well as customers' needs. Business Continuity Management (BCM) is a process aiming at managing risks, emergencies, and recovery plans of an organization during a crisis and ensuring the returning to the normal

business operations [5]. A Business Continuity Plan (BCP) [6] specifies the methodologies and procedures required to backup and recover every functional units of the business.

Traditionally, BCP focuses mainly on the analysis of IT infrastructures and does not consider other aspects of the business such as business-process and business-objective [7,8]. For instance, in a e-Shopping scenario, where the main business-objective is *selling items to customers*, the continuity of business-objective might depend not only from the IT infrastructures (e.g., inventory servers, firewall, payment servers, and authentication servers), but also from the operational-level of the organization, such as *delayed payment services* or even more higher level, such as *existence of new competitors*.

In this paper, we propose a framework to support the analysis of business continuity from a socio-technical perspective. Essentially, we extend our previous work on risk analysis [9] with the light of the Time Dependency and Recovery (TDR) model [8]. Our previous framework is extended in order to analyze the business-objectives, to realize them at more operational level (business process [10] or tasks) and, finally, to identify the required artifacts to execute the processes. To model dependencies among assets (objectives, processes, artifacts), we adopt the time-dependency relation from the TDR model. This proposed framework intends to assists analysts in: 1) analyzing assets, 2) defining additional measures to fulfill the stakeholders' target, and 3) defining the most cost-effective mitigation plans.

The remaining paper is organized as follows. Next we present a running example, the *Loan Originating Process (LOP)* of a bank (§2). We then introduce the modeling framework (§3) that extends our previous Goal-Risk framework with the TDR model, and the analysis processes supported by the framework itself (§4). Then, we apply the framework to the LOP case study to evaluate our proposal (§5), and, finally, we discuss related works (§6) and conclude the paper (§7).

2 Running Example

The case study that we use in this paper is originated within the European project SERENITY.[1] It focuses on a typical *Loan Origination Process* (LOP) that starts by receiving a loan application and ends, possibly, with the loan approval. Essentially, a Loan Department within a bank is responsible to accept loan applications, handle the applications, and ensure the loan repayment. These objectives are operationalized through a set of business processes. For instance, once the bank receives a loan application, it starts the handling process verifying the data and calculating the credit score. The score is assessed either internally (in-house assessment) or by an external party (Credit Bureau). Afterward, the bank defines the loan schema, namely defining the loan cap and its interest. In this example, we assume it is always the case that the customer agrees with the loan schema proposed by the bank. Surely, the bank is also interested in ensuring the repayment of the loan.

Uncertain events (i.e., threats, un/intentional events, incidents, risks) may affect the availability of assets. For instance, events like computer virus outbreak, database

[1] http://www.serenity-project.org/

failure, the outage of national identity service are considered as disruptions for the loan department. Essentially, these disruptions are hard, or impossible, to avoid, but they might be still acceptable if their effects vanish after an acceptable period (called Maximum Tolerable Period of Disruption-MTPD). For an example, the goal of receiving loan is still satisfied though it is disrupted for 2 hours. To maintain the MTPD, all responsible stakeholders establish a contingency plan in case their assets are disrupted. The plan, typically, consists of the Recovery Time Objectives (RTOs) that represent the recovery time of assets. For instance, the IT department ensures that the database of loan application system will be recovered within 1 hour after the disruption. For any set of uncertain events, analysts should assess the sufficiency of RTOs to meet the MTPD. In the case of insufficiency, additional measures need to be introduced. Moreover, these additions should be analyzed carefully before their adoption because they introduce additional costs, and very often they introduce other kind of problems to the system.

3 Modeling Framework

To assess BCP, we need to identify and analyze any related assets that are involved in the business. To this extend, we use the Tropos Goal-Risk (GR) framework [9] to analyze risk and Time Dependency and Recovery model [8] to capture interdependencies among assets. A Business Continuity Plan (BCP) is defined in terms of a set of RTOs for all assets. Ideally, it must satisfy the MTPD of business objectives required by stakeholders.

In the following subsections, we explain the underlain framework (TDR model), which captures time dependencies among assets. Afterward, we present the extension of the GR framework for analyzing the Business Continuity in an organization, and also the process to develop a GR model.

3.1 Time Dependency and Recovery Model

The TDR model allows us to model the interdependencies between assets in realizing business objectives.

Definition 1. *A TDR model is a pair* $\langle N, \rightarrow \rangle$ *where* N *is a set of nodes (assets) and* $\rightarrow \subseteq N \times N$ *represents inter-dependency relations between nodes that is tolerable for a given time* t*.*

For example, in Fig. 1, the task entry loan application by Bank Employee *(*T_{02}*)* requires the resource secure desktop client *(*R_{02}*)*. We depict this as $T_{02} \overset{15'}{\longmapsto} R_{02}$ that refers to T_{02} will be not available if R_{02} is unavailable for more than 15 *time unit* (in this paper, we use *minute* as a default time unit). Dash-lines refer to the concept of OR dependency, for instance G_{02} can depend either on T_{01} or T_{02} .

Using the reasoning framework proposed in [8], we can assess the sufficiency of RTOs for all assets against the MTPD of business objectives. Moreover, the proposed tool is able to calculate the Maximum Recovery Time (MRT) of each asset. If all RTOs of assets are less-or-equal of the MRTs, then the continuity of business objectives is guaranteed. Contrarily, the continuity of business might be disrupted. In the case of

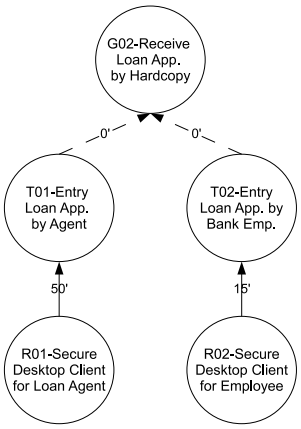


Fig. 1. The TDR Model

RTOs have not been defined, we may always use the MRT as threshold for RTO in order to guarantee the business continuity.

3.2 The Goal-Risk Framework

To model and assess BCPs, we need to analyze 1) business objectives and their realizations (process and artifacts), 2) interdependencies among assets, and 3) the level of risk that threats business objectives, directly or indirectly. However, the "original" GR framework [11] is able to deal with 1 and 3, while the TDR model focuses more on 2. The idea here is to adapt the notion of inter-dependency relation from the TDR model. Thus, the GR framework is able to capture the assets in an organization and is able to model and analyze the BCP.

The Tropos Goal Risk (GR) framework introduced in [11] (more details in [9]) adopts the idea of three layers analysis from Defect Detection Prevention (DDP) [12]. It consists of three conceptual layers – asset, event, and treatment (as depicted in Fig. 2) – to analyze the risk of uncertain events over organizations' strategies. The **asset layer** analyzes business objectives of the stakeholders and their realizations (i.e., processes and artifacts), whereas the **event layer** captures uncertain events along their impacts to the asset layer and the **treatment layer** models treatments to be adopted in order to mitigate risks.

Definition 2. *A GR model is a set of tuple* $\langle \mathcal{N}, \mathcal{R}, \mathcal{I} \rangle$*, where:*

- $\mathcal{N}$ *is a set of nodes of three types:* goals*,* tasks*,* resources*, and* events*;*
- $\mathcal{R}$ *is represented as* $(N_1, \ldots, N_n) \xrightarrow{r} M$*, where* $N_i \in \mathcal{N}$*,* $M \in (\mathcal{N} \cup \mathcal{I})$*, and* r *is the type of the relation.* $N_1, \ldots, N_n$ *are called* source nodes *and* M *is the* target node*.* r *consists of* AND/OR-decomposition*,* contribution*, and* alleviation*,* means-end*, and* needed-by[2]*;*

[2] This is a new kind of relation that was not used in the original GR framework

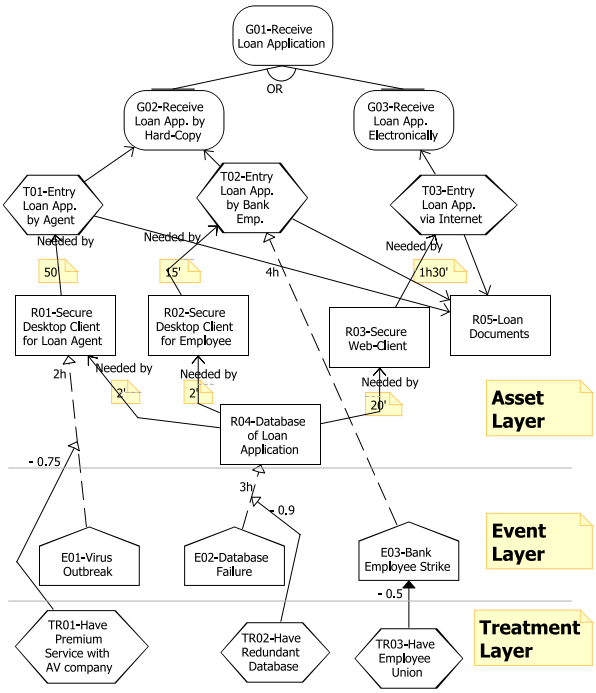


Fig. 2. The Extended GR Model

– $\mathcal{I} \subseteq \mathcal{E} \times (\mathcal{N} \setminus \mathcal{E})$ *is a special type of relation, called impact relation. It relates events* ($\mathcal{E} \subseteq \mathcal{N}$) *with other constructs* ($\mathcal{N} \setminus \mathcal{E}$) *representing the severity of events toward the asset layer.*

Goals (depicted as ovals in Fig. 2) represent the objectives that actors intend to achieve. *Tasks* (hexagons) are course of actions used to achieve goals or treat events. Tasks might need *resources* (rectangles) during their execution or even produce resources. To avoid confusion between tasks for achieving goals and tasks for mitigating risk, from now on we name the former as tasks and the latter as *treatments*, respectively. To model a situation where a task is a means to achieve the end-a goal, we adopt the Tropos [13] *means-end* relation (line arrow), and similarly for the task that produces a resource. So for example, the tasks entry loan application by agent (T_{01}) and entry loan application by bank employment (T_{02}) are means to achieve the goal receive application by hard-copy (G_{02}). Moreover, either T_{01} or T_{02} produce the resource of loan documents (R_{05}), which later might be used in other processes.

To analyze BCP, a GR model needs also to capture assets dependencies. We introduce the *needed-by* relation, adapted form the TDR model, to model a task that needs a particular resource, a resource that needs another resource or a task that needs another task. This type of relation is annotated with time, which represents the maximum disruption period that is tolerable by dependent assets (we use minutes as default time

unit). For example, Secure desktop client for Loan Agent *(R_{01})* is needed by the task T_{01} (the time-dependency is 20 minutes) and R_{01} requires to access database of loan applications *(R_{04})* (the time-dependency is 2 minutes). The disruption of R_{01} will not result in the failure of T_{01} for more than 20 minutes. For computing the MRT, a GR model (2) uses the proposed reasoner in [8] since we can develop the corresponding TDR model (1) following the rules described in the Section 4.

The fulfillment of goals (also the execution of tasks and the provision of resources) might be disrupted by the occurrence of uncertain events (pentagons). Here, an event is characterized into two attributes: likelihood (λ) and its consequences [14].[3] To simplify the calculation the *likelihood* is represented in terms of the number of occurrences within a specific period (in this framework, the time period is a year) judged by experts.[4] For instance, if an expert judges that the likelihood of an event is 0.1, then it implies that the event took place once in 10 years. To model consequences, we use the *impact* relations (dash-line with hallow arrow).

Possible treatments are introduced in the treatment layers. They are aiming at mitigating risks (events with negative impact). Moreover, with the help of a CASE tool [5], analysts can define, which treatments should be adopted to achieve the acceptable risk level.

3.3 Modeling Process

The modeling process of a GR model starts from the **asset layer**, which consists of objectives, processes, and artifacts. We initially identify all business objectives (goals) of stakeholders and then we refine them by iterative decompositions. For example, we identify that stakeholders have two top-goals: receive loan application *(G_{01})* and handle loan application *(G_{04})*. Then, goal G_{01} is OR-decomposed into receive loan application by hard-copy *(G_{02})* or receive loan application electronically *(G_{03})*. The refinement process continues until each leaf-goal is tangible, that is there exists at least a task to fulfill it. As soon as analyst identifies the processes/tasks (the operation level of the asset layer) that realize the business objectives, the modeling process continues with the refinement of tasks using AND/OR decomposition. The process stops when each leaf-task is an atomic activity that cannot be anymore broken down in sub activities [10]. Finally, we analyze whether there are necessary artifacts/resources (e.g., T_{01} requires R_{01}) to execute tasks (the artifact level of the asset layer). Some of resources may require other resources (e.g., R_{01} requires R_{04}) or produced by the execution of tasks (e.g., T_{01} produces R_{05}).

The fulfillment of business objectives might be disrupted by the occurrence of uncertain external events. Essentially, in **the event layer** we identify negative events (i.e., threats, un/intentional events, incidents) that disrupt business objectives direct or indirectly (by disrupting the supporting assets). For instance, the resource secure desktop

[3] In this paper, we consider only events with negative consequences (i.e., risks, threats, incidents).

[4] The model allows us to represent the likelihood in terms of *Probability Distribution Function* for a better result (i.e., precision), but it requires more complex mathematical computation

[5] `http://sesa.dit.unitn.it/sistar_tool`

client for loan agent *(R_{01})* might be disrupted by the occurrence of virus outbreak *(E_{01})*. This event will cause 2 hours of unavailability for R_{01} . Taxonomy-based approaches, such as Computer Program Flaws [15], Faults [16], can be used to identify this class of events related to the software systems. For identifying events in other domains (e.g., management, financial), analysts should conduct the interviews to the related stakeholders or the domain experts. However, the availability of resources is not sufficient to guarantee the continuity of business objectives. There could be circumstances where the disruption is introduced from the process level. For example, the task entry loan application by bank employee *(T_{02})* can be unavailable for 4 hours because of the occurrence of event bank employee strike *(E_{03})*. To identify risks at this level, we can use organizational-driven [17,18] and again taxonomy-based [19] approaches.

Suppose the bank intends, also, to satisfy the goal ensure loan repayment. This objective can be realized in two different ways (processes): 1) assessing the credit score and 2) underwrite the loan according to the credit score. Though, the bank is able to carry on both processes to ensure the repayment of the loan, the risk of a economic crisis may still disrupt the business objective. For this type of events, obstacle approach [20] can be used.

We recommend analysts to start the event identification process from the artifact level and then move up to the process and objective level. In this manner, we prevent the spurious identification of an event's impact. For example, the event virus outbreak *(E_{01})* might be modeled to impact the goal receive loan application *(G_{01})*. However, this is not correct because actually E_{01} obstructs R_{01} that is used to fulfill G_{01} . In other words, if an event disrupts a resource, then certainly it will also produce a similar effect to tasks that use such a resource and consequently this will affect goals that the tasks are supposed to satisfy. Conversely, in the case of the event economic crisis and the goal repayment of the loan, the event does not obstruct any task or any resource that are realized the goal. Identified events are refined using again decomposition relations until all leaf-event are assessable.

Once the strategic and event layers have been analyzed, we identify and analyze the countermeasures that might be adopted to mitigate risk in **the treatment layer**. To mitigate risks, treatments can operate in two ways: reducing likelihood and/or reducing Time-Period of Disruption (TPD). To reduce the likelihood, we use the *contribution* (depicted as line with filled-arrow) with the annotation ($[-1, 0)$) indicating the extent of likelihood reduction. For instance, the treatment have employee union *(TR_{03})* mitigates to 50% the likelihood of the event bank employee strike *(E_{03})*. It is presumably because the union may intermediate the conflict between employees and employers. However, we use the *alleviation* relation (depicted as line with hallow-arrow) to capture the mitigation of risk impact (in this context is the reduction of TPD). For instance, the treatment have redundant database *(TR_{02})* reduces 0.9 of the TPD caused by the event database failure *(E_{02})*.

Summing up, we have revisited the semantics of relations in the GR framework to reason about business continuity. For instance, in [11] the GR model cannot model the time-dependency among the constructs. Moreover, a impact relation, initially, represents how much evidence (satisfaction and denial) is propagated to the asset layer once

an event occurs. To model "disruption", we need to revisited the semantic of this relation. In this case, an impact relation depicts how long is the disruption once an event occurs. By means of this model, ones can reason about the sufficiency of existing BCP, in terms of RTO, to meet the MTPD. The following section, we present the analysis supported by the model.

4 Analysis Process

Once we have the extended GR model we can analyze the continuity of the business objectives performing two different kinds of analysis.

- *Treatments Analysis*, intended to elicit all possible sets of treatments that are able to mitigate the risk until the acceptable level. Analysts will choose the most adequate mitigation to introduce following some criteria (e.g., additional costs, possible side-effects).
- *Cost-Benefit Analysis*, aiming at identifying the most cost-effective treatments to reduce the loss introduced by business discontinuity. This analysis is useful when there is no possible set of treatments that is able to reduce the level of risk until the acceptable level. In this case, analysts typically choose the most cost-effective set of treatments.

Inputs for both analyses are:

1. A multi-layers model (e.g., Fig. 2 and Fig. 4);
2. Acceptable risk, represented in terms of pairs Maximum Time Period of Disruption (MTPD) and Maximum Likelihood (Max.λ) of disruption for each top goal (e.g., $MTPD(G_{01}) = 60$ minutes - Max.$\lambda(G_{01}) = 2$, $MTPD(G_{04}) = 120$ minutes- Max.$\lambda(G_{04}) = 2$);
3. "Significant" business objectives, which are defined as top-level goals and other subgoals that the stakeholders believe to be important for the organization. For each of these goals, we specifies its utility for the organization [6] (e.g., $Utility(G_{01}) = 80$, $Utility(G_{02}) = 50$);
4. Likelihood of events (e.g., $\lambda(E_{01}) = 12$, $\lambda(E_{03}) = 3$);
5. Treatments costs (e.g., $Cost(TR_{01})$=200, $Cost(TR_{02})$=70).

Definition 3. *For any given Multi-layers model $\langle \mathcal{N}, \mathcal{R}, \mathcal{I} \rangle$, we build a TDR model $\langle N, \rightarrow \rangle$, where:*

- *N is $\mathcal{N}$ in the asset layer;*

[6] We quantify the utility in the range $[0, 100]$. Conceptually, the notion of utility and value are different as indicated in literature about *expected utility* and *expected value* [21]. To assess the utility of an asset, one can assess it by summing up all the values generated by the assets. For instance, a server may have a value not more than 10000, but it may have utility much more beyond its value.

– $\rightarrow$ *is constructed from* $\mathcal{R}((N_1, \ldots, N_n) \stackrel{r}{\longmapsto} M)$ *where* $N_1, \ldots, N_i, M \in \mathcal{N}$ *in the asset layer*

$$\rightarrow = \bigcup_{\mathcal{R}} \begin{cases} N \stackrel{t}{\longmapsto} M, & \text{if } r = \textit{needed-by}^7; \\ N \stackrel{0}{\longmapsto} M, & \text{if } r = \textit{means-end} \wedge \textit{M is a goal} \wedge \textit{N is a task}; \\ M \stackrel{0}{\longmapsto} N, & \text{if } r = \textit{means-end} \wedge \textit{M is a resource} \wedge \textit{N is a task}; \\ \bigcup_{i=1\ldots n}(N_i \stackrel{0}{\longmapsto} M), & \text{if } r = \textit{decomposition}. \end{cases}$$

Compare Fig. 1 and Fig. 2 to have an idea of the correspondence between a TDR model and an Extended GR model. Given a TDR model $\langle N, \rightarrow \rangle$, for each $n \in N$, the MRT ($mrt(n)$) is calculated as follow [8]:

$$mrt(n) = \begin{cases} MTPD_n, & \text{if } N \text{ is a top-goal;} \\ min\{mrt(m) + t) | n \stackrel{t}{\longmapsto} m\}, & \text{otherwise.} \end{cases}$$

4.1 Treatment Analysis

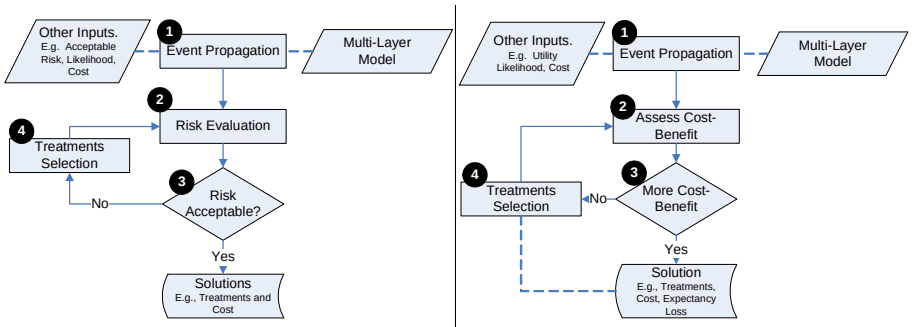
Treatments analysis is represented step-by-step in Fig. 3(a). (Step 1) Risks – likelihood and consequences – of events are propagated throughout the model. (Step 2) We evaluate whether it is possible to satisfy all top goals with a risk under given values. This is done looking at how much the likelihood of top-goals and how long for they will be disrupted. If the risk is unacceptable (Step 3), then we refine the model introducing treatments. In this framework, we adopt the algorithm *Find Treatments* proposed in [9] to identify the necessary treatments. Essentially, the algorithm is an adaptation of the *greedy search algorithm* [22] that aims at suppressing the increase of costs because of new treatments. If the TPD of top-goals is not acceptable (TPD greater than MDTP), then the algorithm will propose treatments connected by alleviation relations. If the TPD is equal to MTPD, then it is acceptable if it occurs less-or-equal than Max.λ, otherwise the algorithm will propose the treatments connected by contribution relations to the event layer (Step 4). Notice in the worst case, this process will explore all possible subsets of treatments (i.e., $2^{N(treatments)} - 1$), which hardly will happen in practice. Finally, we possibly obtain different solutions (a solution consists of several treatments) that satisfy the acceptable risk and cost, and then we decide on the bases of criteria such as cost, stakeholders' preference, company culture, etc. which solution to implement.

4.2 Cost-Benefit Analysis

Cost-benefit analysis is useful when analysts cannot find any possible composition of treatments to mitigate the risk until the acceptable level. This analysis is aiming at finding the most advantageous (i.e., cost effective) solution. The notion of advantageous (ADV) is represented in terms of the ratio between benefit and cost (1) [8], while benefit

[7] t is the time-dependency in a *needed-by* relation

[8] Analysts must adopt at least a treatment to mitigate risk and therefore the $Cost$ cannot be 0



(a) Treatment (b) Cost-Benefit

Fig. 3. Analysis Process

is modeled as an inverse function of the loss expectancy - LE -(2)[9].

$$ADV(S) = \frac{1}{\sum_{G \in significant\text{-}goals} LE(G) \times Cost(S)} \quad (1)$$

$$LE(G) = [\![\lambda(G) - Max.\lambda(G)]\!] \times Utility(G) \times [\![TPD(G) - MTPD(G)]\!] \quad (2)$$

Essentially, the loss is introduced when the TPD is greater than MTPD and it happens more often than Max.λ. In this framework, the loss expectancy is calculated as multiplication of the likelihood distance, the utility of the goal, and the overhead of disruption period.

The overall process of cost-benefit analysis is depicted in Fig. 3(b). (Step 1) a set of treatments is selected, and the loss expectancy of every significant goals and the total cost are calculated to obtain the ADV (Step 2). This process continues exploring every possible combination of treatments (Step 3). Moreover, the notion of cost-benefit might be enriched by considering other factors (e.g., time of implementation, intangible values) besides only loss-expectancy and cost. Notice this process is an exhaustive process that requires to explore all possible subset of treatments. However, some optimization can be taken to reduce the possible search space. For instance, the algorithm records the most cost-effective solution ignoring the branch of search space, which is less beneficial than the recorded solution. Finally, (Step 4) the result of this process is only a solution that theoretically, based on the equation (1), is the most cost-effective solution. Typically, this type of solution would be easy to get an approval by the stakeholders because it proposes the set of treatments, which is the most cost-effective. Moreover, this analysis can be used, in conjunction with the treatment analysis, to evaluate among proposed solutions.

5 Validation through an Example in Large

To evaluate our approach and its implementation, we ran a number of experiments with the *Loan Origination Process* case study that is a simplification of SERENITY

[9] the function "$[\![x]\!]$" never results a value lower than 0. E.g., $[\![5]\!] = 5$, $[\![-2]\!] = 0$, $[\![-0.002]\!] = 0$

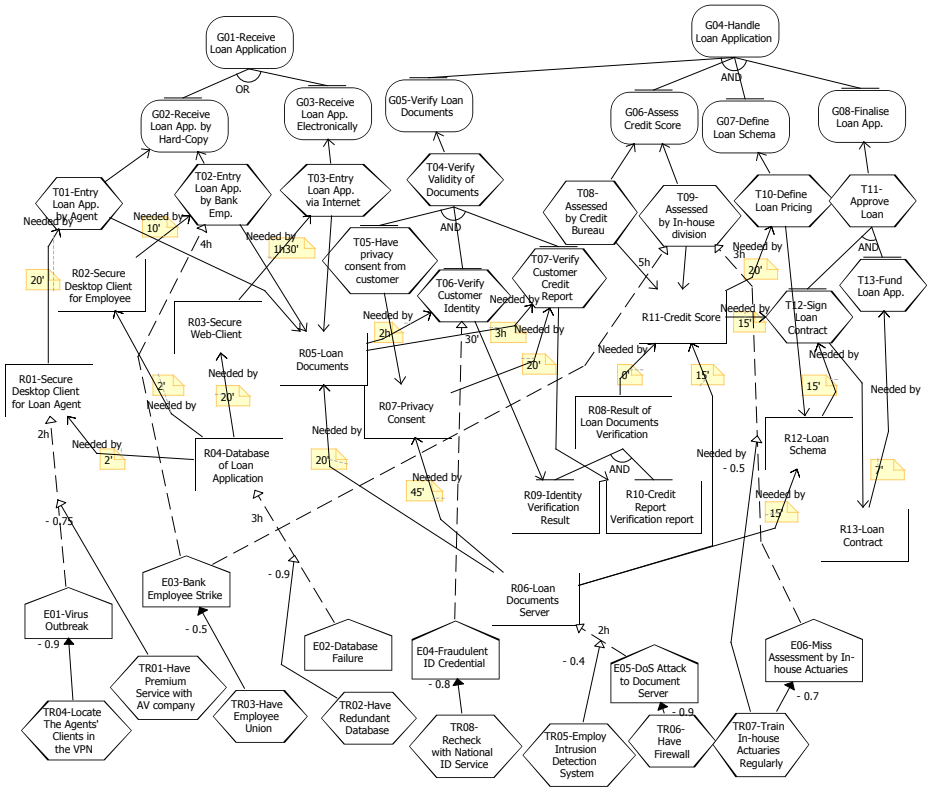


Fig. 4. The Model for Assessing the BCP of Loan Originating Process

e-Business scenario [23]. As illustrated in Fig. 4, let consider two top goals for the bank: receive loan application *(G_{01})* and handle loan application *(G_{04})*. Suppose stakeholders expressed their acceptable risks (i.e., MTPD, Max.λ) for the two goals as indicate in Table 1. For a given MTPD, we compute the MRT of every asset (indicated as the number at upper-left every constructs in Fig. 4) required to satisfy the MTPD. Suppose also, stakeholders argue about the importance of subgoal G_{02}, that can endanger the image of the organization in case it will not be satisfied (even if G_{01} is satisfied). We quantified the G_{02} utility as 50, which is slightly lesser than the utility for G_{01} ($Utility(G_{01}) = 80$). Differently, goal G_{04} $Utility(G_{04}) = 40$) results being less important than G_{01} and G_{02} since its failure will not be visible outside of the organization.

Given these inputs, in Table 3 we see how risks disrupt the business continuity. For instance, R_{01} should have at most 2 times of 80 minutes of disruption (MRT) in one year. Unfortunately, the impact of E_{01} results in 12 times of 2 hours disruption, which is unacceptable. However, the assets of T_{02}, T_{09}, T_{06} are not at risk because either they occurs less than 2 times a year or their disruption is less than their MRT. To mitigate

Goals	MTPD(G)	Max. $\lambda(G)$	$Utility(G)$
G01 Receive Loan Application	60	2	80
G04 Handle Loan Application	120	2	40
G02 Receive Loan App. by Hard-Copy			50

Table 1. The Inputs of Top Goals and "Significant" Goals

Treatment	Cost	S_1	S_2	S_3	S_4	S_5
TR01 Have Premium Service with AV company	200	X			X	X
TR02 Have Redundant Database	50	X	X	X	X	X
TR03 Have Employee Union	100					
TR04 Locate The Agents' Clients in the VPN	90		X	X		
TR05 Employ Intrusion Detection System	30	X		X		X
TR06 Have Firewall	10		X		X	X
TR07 Train In-house Actuaries Regularly	70	X	X	X	X	X
TR08 Recheck with National ID Service	40					
Total Cost		350	220	240	330	360

Table 2. Total Cost of Possible Treatments

Event-Src	Target	MRT	Init	S_1	S_2	S_3	S_4	S_5
E01 Virus Outbreak	R01	2-80'	12-2h	12-30'	1.2-2h	1.2-2h	12-30'	1.2-30'
E02 Database Failure	R04	2-72'	10-3h	10-18'	10-18'	10-18'	10-18'	10-18'
E03 Bank Employee Strike	T02	2-1h	2-4h	2-4h	2-4h	2-4h	2-4h	2-4h
E03 Bank Employee Strike	T09	2-2h	2-5h	2-5h	2-5h	2-5h	2-5h	2-5h
E04 Fraudulent ID Credential	T06	2-2h	24-30'	24-30'	24-30'	24-30'	24-30'	24-30'
E05 DoS Attack to Doc. Server	R06	2-80'	20-2h	20-72'	2-2h	20-72'	2-2h	2-72'
E06 Miss Ass. In-house Actuaries	T09	2-2h	4-3h	1.2-1.5h	1.2-1.5h	1.2-1.5h	1.2-1.5h	1.2-1.5h
Total Cost			350	220	240	330	360	

Table 3. Risks in The LOP scenario Initial and After Treatments Adoption

such risk, treatment analysis enumerates 81 possible solutions (i.e., sets of treatments) that can satisfy the stakeholders' inputs. For the sake of simplicity, we concentrate only on five of them, namely $S_1 - S_5$ as indicated in Table 2.

From Table 3, we can observe that the MRT of R_{06} is 80 minutes for 2 times/year. However, with S_2 the event E_{05} is mitigated into 2 hours for 2 times/year to R_{06} , which is acceptable by the stakeholders because the likelihood of the disruption is not exceeded. However, S_5, which includes treatment TR_{05} , results in 72 minutes for 2 times/year. It implies the business is never discontinued because the TR_{05} can be recovered before the disruption appears in the business level. Each solution has a different cost and also a different impact on the reduction of risk, as presented in Table 3. Notice that all solutions ($S_1 - S_5$) produce an acceptable level of risk, but S_2 results being the cheapest solution. However, S_3 can be also a good candidate since it can reduce, further, the outage-period of R_{06} from 2 hours to 72 minutes with only a bit higher cost. Decision about S_2 or S_3 is now responsibility of analysts, they have to evaluate what is better for the organization.

To show the cost-benefit analysis, we suppose now that stakeholders are more risk averse than in the previous case. MTPD for goals G_{01} and G_{04} are reduced to 2 and 50 minutes, respectively. Consequently, the new MTPD will results in shorter MRT for each asset. Unfortunately, in this case there is no possible combination of treatments

that can reduce the risk until the acceptable level. In this situation, the analyst might simply ignore this fact and accept the risk per se, or consider to adopt the the most beneficial solution.

	S_1	S_2	S_3	S_4	S_5
Disruption after Treatments					
R01	12-8'			12-8'	
R04	10-4'	10-4'	10-4'	10-4'	10-4'
R06	20-7'		20-7'		
Results					
Cost	350	220	240	330	360
LE	21920	4800	10400	16320	4800
ADV (in 10^{-7})	1.30344	9.4697	4.00641	1.8568	5.78704

Table 4. ADV of Possible Treatments in the LOP scenario

Notice in Table 3, the asset of T_{02} and T_{09} results in an acceptable disruption because it happens only twice a year. Though in this setting the MRT of T_{06} is much smaller (e.g., $MRT(T_{06}) = 50'$), the recovery time of T_{06} is much smaller (i.e., $30'$) therefore T_{06} cannot caused unacceptable disruption. Conversely, with S_1 the system still suffers 12 times/year an outage of 30 minutes for R_{01} where the MRT of R_{01} is 22 minutes. In other words, the system is discontinued for 8 minutes, 12 times/year (see Table 4 for the complete ones). Consequently, these outages will introduce a loss (expectancy) for G_{01} , G_{02} , and G_{04} , as define in equation (2). For instance, in Table 4 the resulting loss expectancy for S_1 is 21920 with a cost of 350. Looking at the table, S_2 results the most cost-effective solution, the lowest level of LE and the cheapest cost.

To summing up, this section has presented how this approach works in two settings: 1) resulting a set of countermeasures that need to be introduced to ensure the business continuity of an organization and 2) to find the most cost-effective set of treatments to maintain the business continuity. This approach does not require very precise inputs (e.g., likelihood, time-dependency, etc.). However, we recommend to analysts to use the worst possible scenarios while assessing the inputs, though it means "overshooting risks".

6 Related Work

KAOS [20,24], a goal-oriented requirements engineering methodology, has been proposed aiming at identifying not only *what* and *how* aspect of goals but also *why*, *who*, and *when*. Moreover, KAOS introduces also the concept of *obstacles* [20] and *anti-goal* [24], which can be seen as boundaries in goal analysis. Those two concepts can be used to identify the top-events that may threaten the asset layer of a GR model. Moreover, the refinement of obstacles and anti-goals are compatible of the decomposition of an event.

Liu et al. [25] propose a methodological framework for security requirements analysis based on *i**. They use the NFR framework [26] to support the formal analysis of

threats, vulnerabilities, and countermeasures. This framework captures more details of a malicious events occurs by identifying who is the attacker, what are the vulnerabilities, and what countermeasures should be taken. In our work, we do not distinguish between a disruption due to malicious or non-malicious intents.

Moreover, the works, namely Fault Tree Analysis (FTA) [27] or *attack tree* [28], have similar representation with events in the multi-layer model. Those works capture and analyze the events that may harm the system. Therefore, ones may replace the event layer with those works because of familiarity reason. Notice, those works require objective-quantitative data that can be obtained by recording past experiences.

Approaches like Multi-Attribute Risk Assessment (MARA) [29] can improve the risk assessment process by considering multi-attributes. Many factors like reliable, available, safety and confidentiality can result critical for a system and each of them has its own risk value. This introduces the need for the analyst to find the right trade-off among these factors. In this work, we only assess the recoverability property that is part of the availability. Our results in assessing the recoverability of the system can be useful as one of the input to perform MARA.

Electronic Data Processor (EDP) Audit shares many commonalities with the work in Business Continuity Management. Essentially, the EDP Audit is mirror the activity of business audit [30]. It is a process collecting evidence to determine whether IS systems protect assets, maintain the data integrity, achieve the goals of organization effectively, and consume resources efficiently [31]. To achieve this end, auditors should ensure that the EDP contingency plan is sufficient and has been in place. In this domain, our framework may assist the auditors to analyze the sufficiency of the plan (i.e., RTO).

Finally, approaches on business process modeling, such as Business Process Modeling Notation [32], declarative business process [33], might be useful to structure the process level of the asset layer. It is useful to improve the precision of inter-dependency analysis among assets.

7 Concluding Remarks

In this paper, we have presented a comprehensive framework to analyze the business continuity of an organization. The framework models all levels of assets (e.g., objective, process, and artifact) that may be involved in the continuity of the business. In order to guarantee the continuity of business under uncertainty (e.g., incidents, attacks, human-errors, hardware-failures), we need to introduce a set of treatments to mitigate risks. The proposed framework, allows the analysts to explore and analyze all possible sets of treatments that can be introduced to mitigate the risk (severity or likelihood) of these events. Moreover, the framework also proposes cost-benefit analysis that allows the analyst to select the most cost-effective treatments.

As future work, we intend to introduce more precise description of processes and artifacts in the asset layer by means of more expressive languages (e.g., BPMN, ADL). Moreover, we plan to do more works in order to increase the accuracy of the BCP assessment and its usability. We also intend extending the analysis to a multi-actor environment, where an actor may depend on other actors and they may dis/trust each

other. It is also interesting to explore BCP in organization where business objectives and activities are outsourced to other parties.

However, we are aware that the continuity/recoverability problem is only one issue of a critical system (i.e., security and dependability properties). Therefore, the continuity of a business is necessary for a secure and dependable system but it is not sufficient. There are other issues, such as confidentiality, that may compromise the system though the continuity of business is still guaranteed.

Acknowledgment

This work has been partly supported by the projects EU-SERENITY and PRIN-MENSA. Thanks to Emmanuele Zambon for the discussion and inputs on this work.

References

1. Mate, J.L., Silva, A., eds.: Requirements Engineering for Sociotechnical Systems. Information Science Pub, Hershey, PA (2005)
2. Neumann, P.G.: RISKS-LIST: RISKS-FORUM Digest. http://catless.ncl.ac.uk/Risks/ accessed at 2008-05-27.
3. Basel Committee on Banking Supervision: Basel II: International Convergence of Capital Measurement and Capital Standards: a Revised Framework. http://www.bis.org/ (June 2004)
4. Sarbanes, P., Oxley, M.G.: Public company accounting reform and investor protection act. Washington, DC: Government Printing Office (2002)
5. BSI: Business Continuity Management. BSI 25999-1 (2006)
6. Doughty, K., ed.: Business Continuity Planning Protecting Your Organization's Life. Best practice series. Auerbach, Boca Raton (2001)
7. Lam, W.: Ensuring Business Continuity. IT Professional **4** (2002) 19–25
8. Zambon, E., Bolzoni, D., Etalle, S., Salvato, M.: A Model Supporting Business Continuity Auditing and Planning in Information Systems. In: Proc. of ICIMP'07. (2007) 33
9. Asnar, Y., Giorgini, P.: Risk Analysis as part of the Requirements Engineering Process. Technical Report DIT-07-014, DIT - University of Trento (March 2007)
10. Weske, M.: Business Process Management: Concepts, Languages, Architectures. Springer-Verlag New York (2007)
11. Asnar, Y., Giorgini, P.: Modelling Risk and Identifying Countermeasures in Organizations. In: Proc. of 1st Int. Workshop on Critical Inform. Infrastructures Sec. (CRITIS '06). Volume 4347 of LNCS., Springer (2006) 55–66
12. Feather, M.S., Cornford, S.L., Hicks, K.A., Johnson, K.R.: Applications of Tool Support for Risk-Informed Requirements Reasoning. Computer Systems Science & Engineering **20**(1) (2005)
13. Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: Tropos: An Agent-Oriented Software Development Methodology. JAAMAS **8**(3) (2004) 203–236
14. ISO/IEC: Risk Management-Vocabulary-Guidelines for Use in Standards. ISO/IEC Guide 73 (2002)
15. Landwehr, C.E., Bull, A.R., McDermott, J.P., Choi, W.S.: A Taxonomy of Computer Program Security Flaws. ACM Comp. Surveys **26**(3) (1994) 211–254
16. Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.E.: Basic Concepts and Taxonomy of Dependable and Secure Computing. TDSC **1**(1) (2004) 11–33

17. Bhuiyan, M., Islam, M., Koliadis, G., Krishna, A., Ghose, A.: Managing business process risk using rich organizational models. In: Computer Software and Applications Conference, 2007. COMPSAC 2007 - Vol. 2. 31st Annual International. Volume 2. (2007) 509–520
18. COSO: Enterprise Risk Management - Integrated Framework. Committee of Sponsoring Organizations of the Treadway Commission. (September 2004)
19. Carr, M.J., Konda, S.L., Monarch, I., Ulrich, F.C., Walker, C.F.: Taxonomy-Based Risk Identification. Technical Report CMU/SEI-93-TR-6, SEI-CMU (June 1993)
20. van Lamsweerde, A., Letier, E.: Handling Obstacles in Goal-Oriented Requirements Engineering. TSE **26**(10) (2000) 978–1005
21. Bernoulli, D.: Exposition of a New Theory on the Measurement of Risk. Econometrica **22** (1954) 23–36 (original 1738).
22. Russell, S.J., Norvig, P.: Artificial Intelligence: A Modern Approach. 2nd edn. Prentice-Hall (2003)
23. Asnar, Y., Bonato, R., Bryl, V., Campagna, L., Dolinar, K., Giorgini, P., Holtmanns, S., Klobucar, T., Lanzi, P., Latanicki, J., Massacci, F., Meduri, V., Porekar, J., Riccucci, C., Saidane, A., Seguran, M., Yautsiukhin, A., Zannone, N.: Security and Privacy Requirements at Organizational Level. Research report A1.D2.1, SERENITY consortium (November 2006) EU-IST-IP 6th Framework Programme - SERENITY 27587.
24. van Lamsweerde, A., Brohez, S., Landtsheer, R.D., Janssens, D.: From System Goals to Intruder Anti-Goals: Attack Generation and Resolution for Security Requirements Engineering. In: Proc. of RHAS'03. (2003)
25. Liu, L., Yu, E.S.K., Mylopoulos, J.: Security and Privacy Requirements Analysis within a Social Setting. In: Proc. of RE'03. (2003) 151–161
26. Chung, L.K., Nixon, B.A., Yu, E., Mylopoulos, J.: Non-Functional Requirements in Software Engineering. Kluwer Academic Publishers (2000)
27. Stamatelatos, M., Vesely, W., Dugan, J., Fragola, J., Minarick, J., Railsback, J.: Fault Tree Handbook with Aerospace Applications. NASA (2002)
28. Schneier, B.: Attack Trees: Modeling Security Threats. Dr. Dobbs Journal **12**(24) (1999) 21–29
29. Butler, S.A.: Security Attribute Evaluation Method: a Cost-Benefit Approach. In: Proc. of ICSE'02, New York, NY, USA, ACM Press (2002) 232–240
30. Bace, R.G.: Intrusion Detection. Sams Publishing (2000)
31. Weber, R.: EDP Auditing. McGraw-Hill (1982)
32. López, H.A., Massacci, F., Zannone, N.: Goal-Equivalent Secure Business Process Reengineering for E-Health. In: Proc. of MOTHIS'07. (2007)
33. Bryl, V., Mello, P., Montali, M., Torroni, P., Zannone, N.: B-Tropos: Agent-Oriented Requirements Engineering Meets Computational Logic for Declarative Business Brocess Modeling and Verification. In: Proc. of CLIMA VIII. (2007)