Goal-Oriented Requirement Engineering Support for Business Continuity Planning

Alvaro E. Arenas[1], Philippe Massonet[2], Christophe Ponsard[2], and Benjamin Aziz[3]

[1] IE Business School, IE University, Madrid, Spain.
alvaro.arenas@ie.edu
[2] CETIC Research Centre, Charleroi, Belgium.
{philippe.massonet,christophe.ponsard}@cetic.be
[3] School of Computing, University of Portsmouth, Portsmouth, U.K.
Benjamin.Aziz@port.ac.uk

Abstract. Business continuity is a key management process that aims to maintain and rapidly recover an organizations key business functions in the face of serious incidents. The resulting business continuity plan must identify the key business functions that must be resilient, define recovery of critical business functions and define contingency measures when recovery is not possible. This paper argues that the process of business continuity planning can be efficiently supported by a goal-oriented requirements engineering approach. The main benefits of a modelling approach include taking a holistic approach when analysing the organisation, providing quality checks and related guidance across in all the elaboration phases, an supporting the generation of the continuity plan from a business continuity model.

1 Introduction

Business continuity management (BCM) is a management process to ensure the continuity of critical business functions in an organization after a business interruption [2]. The potential causes of business interruption include, among others, natural disasters, human errors, utility interruption such as power outages, or malicious threats from outsiders. Business continuity has become a topic of interest to organisations nowadays due to the recognition that any interruption in the continuity of the business for an extended period of time seriously affect the overall viability of the business, which is of paramount importance in today global economy and competitive environment. Simply recovering the business function is not enough; the business needs to resume as quickly and as efficiently as possible. Recovering the business function entails numerous corporate goals such as preservation of the customer base, restore IT systems, ensure cash flow, and maintain corporate image, among others.

There are several approaches for developing BCM, ranging from standards such as the ISO 22301 standard for business continuity management [8], international initiatives such as the European approach to business continuity led by EU Agency for Network and Information Security (ENISA) [6], practical approaches such as the three phases of business continuity planning [5], and academic proposals to continuity management [3, 9, 12]. All these approaches somehow address three interdependent objectives: (i)

Identify major risks of business interruption; (ii) develop a business continuity plan (BCP) to mitigate or reduce the impact of the identified risk; and (iii) train employees and test the plan to ensure that it is effective.

A BCP can be seen as the document that defines the resources, actions, tasks and data required to manage the business recovery process in the event of a business interruption. A BCP seeks to eliminate or reduce the impact of a disaster condition before the condition occurs. A BCP should evolve as the business environment changes and its dependency on technology changes. However, BCM is criticised for no taking a holistic approach when analysing the organisation, and a lack of clear understating of the responsibilities of the BCP [7]. In this paper we advocate for the use of goal-oriented requirements engineering techniques (GORE) [14] to help in the development of a BCP with the aim of overcoming these limitations. We have used ENISA approach as our underlying business continuity approach [6].

This paper shows how goal-oriented requirements engineering concepts and analysis techniques (goal refinement, obstacle refinements) can strongly and systematically support the process to produce a high quality BCP, i.e. addressing relevant risks, identifying the critical assets and addressing them through adequate controls. Our work is more specifically anchored in the KAOS goal-oriented approach [14], although alternatives will be discussed.

This work contributes to the research and practice on business continuity management. On the research side, this study proposes a new approach to BCM, incorporating goal-oriented requirement engineering in the developing of continuity plans. We apply model-based techniques to provide quality assurance in the elaboration process, and to automate the generation/update of a BCP. The resulting document could then be completed by BCM specialists within the organisation. On the practical side, this study provides practitioners with a toolkit to analyse their main continuity requirements, to guide them to address key (risk) issues and to help in the generation of a BCP draft according to the needs of their organisations.

The structure of the paper is as follows. Section 2 presents the business continuity process to produce a business continuity plan addressing the right risks for critical assets through adequate controls, introducing a case study used as running example. Section 3 provides the required background on the KAOS goal-oriented methodology that will be applied in section 4 for providing GORE support to the business continuity process. Section 5 will discuss related work. Finally, section 6 will provide some conclusion and future work.

2 Business Continuity Process

2.1 ENISA Business Continuity Management for SMEs

Our underlying BCM model is the one proposed by ENISA [6], which is based on some elements from the OCTAVE ALLEGRO Risk Assessment Methodology [4] and tailored for the case of small and medium enterprises (SMEs). The ENISA approach consists of four phases, explained below.

Phase 1. Select Risk Profile. In this phase, main risks for the organisation are identified using a predefined set of qualitative criteria. Four risk areas as suggested: legal

and regulatory, reputation and customer confidence, productivity and financial stability risks. Risks in each area are classified as high, medium and low. The output of this phase is an organisation risk profile.

Phase 2. Critical Asset Identification. In this phase, critical business functions are selected based on their relative importance to the organisation. Critical business functions are functions whose interruption will lead to an organisation suffering from serious financial, legal and/or other loss or penalty. For each critical business function, it is identified who is responsible for it and which assets are used in the function. An important step in this phase is the "Asset Continuity Requirements", concerned with the analysis of the continuity requirements of the identified assets. This phase comprises three steps: (i) Business Function Selection; (ii) Asset Type Selection, selecting the assets that each business function requires in order to be delivered; and (iii) Asset Continuity Requirement Analysis, concerned with the analysis of the continuity requirements of the identified assets.

Phase 3. Control Selection. Controls refer to measures defined to control the identified risks. Risk controls can involve the implementation of new polices and standards, physical changes and procedural changes that can reduce or eliminate certain risks within the business. The ENISA approach suggests two categories of controls: (a) organisational continuous controls, which are applicable to the organisation horizontally and are concerned with practices and management procedures; and (b) asset-based continuity controls, which are applicable to particular classes of critical assets. The approach includes a set of pre-defined controls in the form of control cards. This phase comprises three steps: (i) Select Organisational Continuity Controls; (ii) Select Asset-Based Controls; and (iii) Document List of Selected Controls.

Phase 4. Implementation and Management. In this phase, current continuity practices are evaluated and assessed the gaps between these practices and the selected controls. The output of this phase is the BCP.

In order to implement successful BCM within an organisation, it must first be initiated as a project, including well defined project structure, scope, objectives and deliverables. Once the Business Continuity project has been established, and in order to be able to commence development of the suite of BCP, it is essential to understand the organisation with respect to its mission critical activities or services, its organisational structure, roles and stakeholders. The ENISA approach exploit the existence of cards for assets and controls as a way of eliciting continuity requirements. We propose here to enrich phases 2 and 3 of the process with goal-oriented requirements engineering. We will exploit model-based technology to generate a draft of the BCP that would help in producing the final version in phase 4

2.2 Running Case Study

We apply the ENISA approach extended with goal-oriented requirements to a case study presented in [6]. The case refers to a dental equipment supplier based in north England. The company supplies both the equipment as well as their maintenance. Most of the customers have contracts of annual maintenance. In addition a significant percentage of the customers have special contracts for expedited repair in case of equipment breakdown.

These special contracts guarantee a repair of the equipment within the next business day of filing the request when no spare parts replacement is required.

In the case where spare parts need to be replaced then the required maximum time to repair is four business days to allow for the shipment of spare parts from the manufacturer. In general no other special limitations and hard requirements exist for this company. The company employs 8 persons full time including the owner. Financial matters are handled by the owner with the support of the secretary and an external accountant. In addition the IT needs of the company are covered with external support from a local IT expert who is engaged on-demand to resolve problems that may arise or implement new solutions upon request.

3 Goal-Oriented Requirements Engineering

Goal-oriented requirements models are structured into different sub-models: a goal model which is the driving model (the "WHY"), an object model to structure the domain description (the "WHAT"), an agent model to capture responsibilities (the "WHO") and an operation model for specification level (the "HOW" dimension); these models are elaborated using a method like KAOS [13][14].

A goal is a prescriptive statement of intent about some system (existing or to-be) whose satisfaction in general requires the cooperation of some of the agents. Agents are active components, such as humans, devices, legacy software or software-to-be components that play some role towards goal satisfaction. Some agents thus define the software whereas the others define its environment. Goals may refer to services to be provided (functional goals) or to quality of service (non-functional goals). Unlike goals, domain properties are descriptive statements about the environment, such as physical laws, organisational norms or policies, etc.

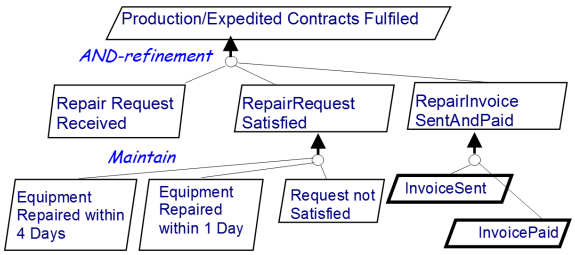

Fig. 1. Business continuity Goal Model Fragment.

Goals are organized in AND/OR refinement hierarchies where higher-level goals are in general strategic, coarse-grained and involve multiple agents whereas lower-level goals are in general technical, fine-grained and involve fewer agents. In such structures, AND-refinement links relate a goal to a set of subgoals (called refinement) possibly conjoined with domain properties; this means that satisfying all subgoals in the refinement is a sufficient condition in the domain for satisfying the goal. OR-refinement links may relate a goal to a set of alternative refinements. Goal refinement ends when every subgoal is realizable by some individual agent assigned to it, that is, expressible

in terms of conditions that are monitorable and controllable by the agent. A requirement and expectations are leaf goals respectively under responsibility of an agent in the software-to-be or the environment.

Goals refer to objects which are structured in models typically represented by UML class diagrams. Objects have states defined by the values of their attributes and associations to other objects. They are passive (entities, associations, events) or active (agents). In the above formalization, *finished* and *invoiceState* are attributes of the *Repair* entity declared in the object model. If the goal *InvoiceSent* is assigned to the *FinanceManager* agent, the latter must be able to monitor the attribute finished and control the attribute *invoice* of *Repair*.

Obstacles anticipate what could go wrong with the system design [15]. An obstacle is a pre-condition for the violation of a goal. Obstacles is the dual concept to goal and like goal then can be refined into sub-obstacles using a AND-OR refinement tree. An obstacle diagram for a given goal is a tree that shows how a root obstacle is refined into sub-obstacles. Example of obstacles for the case study are given in the next section.

Goals can be operationalized into specifications of operations to achieve them. In the scope of this paper, we will not consider the operation level.

4 GORE Business Continuity Process

In this section we show how goal-oriented analysis supports the different phases of the business continuity process described in section 2. Main contributions of goal-oriented analysis are done in phases 2 and 3 of the ENISA approach. Table 1 summarises our business continuity process.

4.1 Select Risk Profile

Risk profile selection is a high level analysis. It targets four areas: legal and regulatory, reputation and customer confidence, productivity and financial stability risks. We assume that the risk profile selection has been done prior to modelling and that it is an input to the modelling process. Those risk areas are modelled as goals/obstacles and refined, risks related to regulation must be modelled as obstacles to goals for regulation and customer satisfaction.

4.2 Critical Assets Identification

Phase 2 involves 3 steps: ”Business Function Selection”, ”Select Asset Types” and ”Asset Continuity Requirements Analysis”. The KAOS goal model has been used to model the first step. In this step critical business functions are identified as Goals because those function directly relate to key organisational objectives. Those can be refined down to atomic critical business operations, providing useful checks and related guidance such as refinement completeness and the existence of responsible agents.

Figure 2 shows how the high level business continuity function ”ProductionExpeditedService ContractsFulfilled” is refined into three main business sub-goals ”RepairRequestReceived”, ”RepairRequestSatisfied”, and ”RepairInvoiceSentAndPaid”. The

Table 1. Mapping between Business Continuity and GORE concepts

BCM Step	GORE Model	Comment
PHASE 1: risk profiles selection		
Selection in available profiles	Generic strategic goal and risk driving the next phase	Specific model pattern for legal/regulatory/-customer confidence can be used
PHASE 2. Critical Asset Identification		
Business Function	Selection Goal model	Critical business functions are modelled as goals and refined into sub-goals
Asset Type Selection	Object model	Assets used in critical business functions are modelled as entities
Asset Continuity Requirements	Obstacle model	Identify obstacle to critical business functions
PHASE 3. Control Selection		
Select Organisation Controls	Obstacle resolution	Identify new requirements that provide organisational controls
Select Asset-Based Controls	Obstacle resolution	Identify new requirements that provide asset-based controls
PHASE 4. Implementation and management		
Gap between practices and controls	AS-IS vs TO-BE gap analysis	Same model can be used to just highlight the gap
BCP production	Report generation based on GORE model	Model-based generation enable easy update

business critical goal ”RepairRequestSatisfied” covers different cases and thus needs to be refined into three sub-goals ”Equipment RepairedWithin 4 Days”, ”EquipmentRepairedWithin1Day”, and ”EquipmentCannotBeRepaired”.

The second step in phase 2 ”Select Asset Types” has been modelled with the object model capturing all entities and relationships bound to critical business functions.

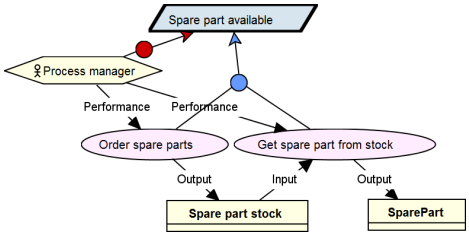

Fig. 2. Agent and Operation Model

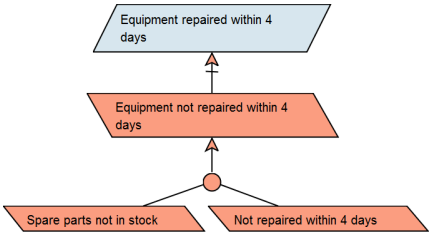

Fig. 3. Obstacle Model

Figure 2 shows the responsibilities in terms of agents, the operations and objects involved. The figure shows that the ”Process Manager” agent is responsible for the requirement ”SparePartAvailable”, and that he can perform two operations to create state transitions to a state where spare parts are available. The operations cover the two

cases where a spare part is available in the ”SparePartStock” and the case where it is not and needs to be ordered via the operation ”OrderSparePart”.

To cover step 3 of phase 2 ”Asset Continuity Requirement Analysis” we have used the KAOS obstacle model. In this model obstacles to critical business functions are identified and refined into sub-obstacles.

Figure 3 illustrates how the risk analysis for business continuity can be modelled using the KAOS concept of Obstacle. In the figure the critical business function that is to repair equipment within 4 days has been captured as a goal. The risk analysis identifies obstacles to the critical business function. This is captured as an obstacle ”Equipment not repaired within 4 days” in the model. This obstacle is in turn refined into two sub-obstacles ”Spare parts not in stock” and ”Not repaired within 4 days”.

4.3 Controls Selection

Phase 3 of the BC analysis involves 3 steps: ”Select Organizational Controls Cards”, ”Select Asset Based Controls Cards” and ”Document List of Selected Controls”. The first two steps are modelled as obstacle resolutions. Obstacle resolution identifies new requirements that provide resolutions to the goal obstacles. Different tactics are available to identify resolutions to the obstacles. Step3 of the phase corresponds to documenting the selecting controls and presenting the rationale for the selection. This documentation is generated from the model using a requirements report generator. This will be detailed in the next subsection.

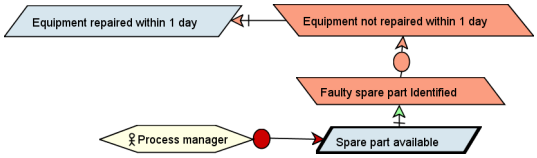

Fig. 4. Obstacle resolutions

Figure 4 shows how business continuity controls are identified to mitigate the risks to the business critical functions. This is captured in the requirements model by introducing a new requirement that resolves the obstacle. The figure shows that the goal ”EquipmentRepairedWithin1Day” is obstructed by the obstacle ”EquipmentNotRepaired-Within1Day”, which is itself refined into obstacle ”FaultySparePartIdentified”. This latter obstacle means that the spare part used to repair the equipment reveals itself to be faulty. This obstacle is resolved by a new requirement ”SparePartAvailable” that describes state transition to a state where a spare part is available to replace the faulty spare part.

4.4 Implementation and Management - Creation of the BCP

For the BCP to be realisable, the proposed controls must be available. For this a specific gap analysis should take place with respect to the existing practices. GORE provide strong support to this because a classic activity is to compare the as-is with to-be

situation. Both model generally share goals but could differ in more specific requirements and operationalisation. Typically some operationalisation could be missing or not achievable in the as-is situation. OR-refinement can be used to capture this in a single model and highlight the gap, e.g. specific controls to be added should be tagged with dedicated system alternative tag like "BCP".

Based on the rich GORE model, generating the BCP is just a matter of querying the appropriate information to feed the right section of the plan. Specific queries can easily be written to generate all the relevant table present in standard BCP template [6] such as: Critical Business Function List, BC Team Responsibility matrix, Business Function Protection Strategy, and Business Function Assets Recovery Actions.

We implemented the described mapping in the Objectiver GORE tool [11] which supports report generation both to text (RTF,ODT) and table format (XLS,ODS) using powerful queries [10]. For example here is the query that will automatically generate the Critical Business Function List presented in Table 2.

```
SELECT a.name AS Dept, g.name AS Function, g.Def AS Definition, g.Pri AS Priority
FROM Assignment AS ass, ass.parent AS g, ass.sons AS s, s.son AS a
ORDER BY g.Priority DESC
```

Table 2. Model-generated Critical Business Function List

Dept	Function	Definition	Priority
Production/ Repair	Equipment repaired within 4 days	When spare parts need to be ordered then four business days is the defined maximum time to repair.	High
Production/ Repair	Equipment repaired within 1 day	The repairs must be performed next business day when no spare parts are required.	High
Finance	RepairInvoiceSent AndPaid	Manage, store and process financial data generated by the commerce of medical equipment and services...	Medium

As the tool also support instance models, it is also possible to use instance level queries and generate tables specifying concrete roles and attribute, such as John Smith should be the $QualityControl$ agent with cell phone +04 65 78 98 00.

5 Related Work

The literature on business continuity dates back to the 1980s. It is intertwined in a multidisciplinary research area bringing together academics and practitioners from several disciplines such as organisational crisis management, information systems, and information and telecommunication technologies.

Most BCM approaches consists of a set of phases, and lack of tool support. For instance, in [3], Botha and Von Solms present a BCM methodology consisting of seven phases (project planning, business impact analysis, business continuity strategies, strategies implementation, continuity training, continuity testing, and continuity maintenance), and following a cyclic implementation approach comprising of four distinct cycles (back-up, disaster recovery, contingency planning, continuity planning). This exemplifies the limitation of current approaches: no taking a holistic approach when analysing

the organisation, and being too prescriptive, which difficult traceability of continuity requirements. Our approach look to overcome these limitations by incorporating requirements engineering in the business continuity management process.

Recent approaches to BCM has concentrated on adding decision support to the continuity process [12] or automating the generation of a BCP [16]. Winkler and Gilani present in [16]] a model-driven approach to generate a BCP using model-transformation chains to connect data across the different phases in BCM. Their approach is closer to ours in the generation of the plan, but our analysis is enriched by incorporating a well-established requirements methodology as it is the case of KAOS [14].

Another model-based approach has been proposed by Zambon [17]. It focuses on assessing and mitigating the risks related to the availability of the IT infrastructure. The starting point is similar as ours: the limitation of current Risk Management methodologies. The narrower scope also enable to consider specific domain properties, especially the dependencies linking the various constituents of the IT infrastructure are taken into account using incidents propagation model. Our approach is a more generic level but can however cope with domain specific reasoning to some extends, i.e. using the available concepts like domain properties, object model, goal and obstacle refinement semantics. However specific model like for incident propagation are beyond our current scope.

In relation to the use of goal-orientation in BCM, Asnar and Giorgini have used Tropos, another GORE methodology to analyse business continuity [1]. Their work is not related to any standard framework and business continuity process model like ENISA as in our case.

6 Conclusions

BC analysis is mostly a document intensive informal process driven by human analysts. The analysis produces a BC plan that aims to operationally guarantee that key business functions are resilient in the face of serious incidents. The BC plan is a document that identifies business critical functions and describes recovery procedures to make them resilient. This paper has investigated how a model-driven approach could be applied to BC analysis. In a model-driven approach the BC plan is derived from the model, thus improving its quality compared to a human-driven semi-informal BC process. The different steps of BC analysis were modelled in terms of a goal-oriented requirements engineering methodology. The risk analysis and the organisation risk profile was modelled as obstacles in an obstacle model. Critical business functions and processes were modelled as goals in goal models. Controls were modelled as obstacle resolutions in a goal model. The paper then showed how a BC plan could be systematically derived from the model. Such a BC plan could be shown to be complete for all obstacles to business critical functions. We argue that using a requirements modelling language provides higher level abstractions for modelling BC concepts.

A limitation in our work is the lack of empirical validation of the proposed approach. This is our next objective, and we are currently working with some European SMEs in the development of their BCP using our approach. Additional future work includes

refining the mapping between BC concepts and RE concepts, and investigating how formalising some BC properties in terms of requirements could enhance the BC plan.

Acknowledgement

This work was partly funded by the SimQRI project (ERANET CORNET nr 1318172).

References

1. Yudistira Asnar and Paolo Giorgini, *Modelling risk and identifying countermeasure in organizations*, Critical Information Infrastructures Security, Springer, 2006, pp. 55–66.
2. Michael Blyth, *Business continuity management: building an effective incident management plan*, John Wiley & Sons, 2009.
3. Jacques Botha and Rossouw Von Solms, *A cyclic approach to business continuity planning*, Information Management & Computer Security **12** (2004), no. 4, 328–337.
4. Richard A Caralli, James F Stevens, Lisa R Young, and William R Wilson, *The octave allegro guidebook, v1. 0*, Software Engineering Institute (2007).
5. Vin D'Amico, *Master the three phases of business continuity planning*, Business Strategy Series **8** (2007), no. 3, 214–220.
6. ENISA, *It business continuity management. an approach for small medium sized organizations*, European Network and Information Security Agency, ENISA Reports (2010).
7. Michael Gallagher, *Business continuity management*, Accountancy Ireland **35** (2003), no. 4, 15–16.
8. BS ISO, *22301, 2012. societal security. business continuity management systems. requirements*, British Standards Institute, London (2012).
9. Jonna Järveläinen, *It incidents and business impacts: Validating a framework for continuity management in information systems*, International journal of information management **33** (2013), no. 3, 583–590.
10. Christophe Ponsard, Robert Darimont, and Arnaud Michot, *Combining Models, Diagrams and Tables for Efficient Requirements Engineering: Lessons Learned from the Industry*, INFORSID 2015, Biarritz, France, June 2015.
11. Respect-IT, *Objectiver Requirements Engineering Tool*, `http://www.respect-it.com`, 2005.
12. Navid Sahebjamnia, SA Torabi, and SA Mansouri, *Integrated business continuity and disaster recovery planning: Towards organizational resilience*, European Journal of Operational Research **242** (2015), no. 1, 261–273.
13. Axel Van Lamsweerde, *Goal-oriented requirements engineering: A guided tour*, Requirements Engineering, 2001. Proceedings. Fifth IEEE International Symposium on, IEEE, 2001, pp. 249–262.
14. _____, *Requirements engineering: from system goals to uml models to software specifications*, Wiley, 2009.
15. Axel Van Lamsweerde and Emmanuel Letier, *Handling obstacles in goal-oriented requirements engineering*, Software Engineering, IEEE Trans. on **26** (2000), no. 10, 978–1005.
16. Ulrich Winkler and Wasif Gilani, *Model-driven framework for business continuity management*, Service Level Agreements for Cloud Computing, Springer, 2011, pp. 227–250.
17. Emmanuele Zambon et al., *Model-based mitigation of availability risks*, Proc. 2nd IEEE/IFIP Int. Workshop on Business-Driven IT Management, Munich, Germany, May 2007.