Model-driven Engineering in Practice: Integrated Performance Decision Support for Process-centric Business Impact Analysis

David Redlich
Lancaster University
Lancaster, UK
mr.redlich@gmail.com

Ulrich Winkler
Queen's University
Belfast, UK
uli.winkler@gmail.com

Thomas Molka
University Manchester
Manchester, UK
thomasmolka@gmail.com

Wasif Gilani
SAP Research Centre
Belfast, UK
wasif.gilani@sap.com

ABSTRACT

Modern businesses and business processes depend on an increasingly interconnected set of resources, which can be affected by external and internal factors at any time. Threats like natural disasters, terrorism, or even power blackouts potentially cause disruptions in an organisation's resource infrastructure which in turn negatively impacts the performance of dependent business processes. In order to assist business analysts dealing with this ever increasing complexity of interdependent business structures a model-driven workbench named Model-Driven Business Impact Analysis (MDBIA) has been developed with the purpose of predicting consequences on the business process level for an organisation in case of disruptions. An already existing Model-Driven Performance Engineering (MDPE) workbench, which originally provided process-centric performance decision support, has been adapted and extended to meet the additional requirements of business impact analysis. The fundamental concepts of the resulting MDBIA workbench, which include the introduction of the applied key models and transformation chain, are presented and evaluated in this paper.

Keywords

model-driven engineering, model-driven performance engineering, business process management, business impact analysis

1. INTRODUCTION

Events like the hurricane Katrina, 9/11, or the tsunami in Fukushima have a measurable impact on our society, in general, and organisations, in particular. These consequences are often of a enormous scale, e.g. through direct and indirect impact of 9/11 nearly 18,000 businesses were disrupted, dislocated, or destroyed [1]. Whether disruptions have a huge impact, like 9/11, or are rather small, like the temporary loss of connection to the Internet, an organisation has to be prepared and act accordingly in order to avoid or minimise financial and reputation losses, or even legal consequences.

Business Impact Analysis (BIA) methodologies examine consequences caused by adverse events. In a direct way, resources, such as facilities or an organisation's IT infrastructure, are impacted. These, in turn, may negatively affect the performance of dependent business processes, which eventually leads to a reduced operability of the organisation. Because of performance analysis of business processes being a discipline of the Business Process Management (BPM), only the directly associated resources against the process activities, for example, employees, machines, etc., are considered. Additional resource infrastructure (IT and facility level resources), vital for the directly associated resources to function, are in BPM generally not taken into account. Thus the business processes and resource infrastructures are commonly regarded as two segregated domains, which makes it difficult to perform a thorough impact analysis. A second emerging issue is that in both domains numerous different modelling methodologies and languages are employed, respectively. Thus, a business impact analyst is required to consolidate all the related and potentially interconnected information.

This paper introduces the MDBIA workbench, which has been developed to combine both domains in a generic fashion and provide decision support for BIA on the business process level. Thereby, the resulting workbench reuses an existent model-driven framework, which offers performance related decision support for business processes, and enhances it with modelling and analysis capabilities for resource infrastructures and possible disruptions. The resulting MDBIA workbench enables to answer questions like: "How is the performance of a business process impacted in case of an occurrence of a specific disruption?".

The remainder of this paper is structured as follows: In the Section 2, essential background information of the two domains BPM and BIA is provided. This is followed by a section describing ***the Model-driven Performance Engi-***

ICPE'14, March 22–26, 2014, Dublin, Ireland.

ACM 978-1-4503-2733-6/14/03$15.00.
http://dx.doi.org/10.1145/2568088.2576797.

neering (MDPE) workbench on which the MDBIA workbench is based. Then in Section 4, the limitations of the MDPE workbench in terms of BIA are discussed in more detail. These limitations are addressed in the next section which introduces the concepts of the MDBIA workbench, the contribution of this paper. The following section is then evaluating the concepts through a qualitative analysis of a test use case. The paper is concluded thereafter in Section 7, in which also potential future work is proposed.

2. BACKGROUND: BPM AND BIA

BPM is defined by van der Aalst [2] as follows: *"Supporting business processes using methods, techniques, and software to design, enact, control, and analyse operational processes involving humans, organisations, applications, documents, and other sources of information."*. This definition is accompanied with a proposed lifecycle comprising four phases: (1) configure, (2) execute, (3) analyse, and (4) decide. Performance related decision support is, however, considered to be part of the fourth step [3].

When looking upon a business from a process-centric point of view one tends to see and model resources in a quite abstract manner. But nonetheless, resources like employees, facilities, or devices are of importance for any running business. If any of these get damaged or inaccessible, the organisation might not be able to carry out individual tasks anymore or, even worse, it becomes completely dysfunctional. Every year, disruptive events resulting from fire, flood, terrorism, or any other external source seriously harm thousands of businesses [4]. But also small and more frequent disruptions like power interruptions, technical failures, or unavailability of an external service can negatively affect businesses if they are not prepared properly.

In order to have your organisation running smoothly even under unusual circumstances one needs to have a better understanding of the functionality and interconnection of resources and in some cases have backup plans in place. Procedures for sustaining necessary business operations while recovering from a considerable disruption are combined in a so called *Business Continuity Plan* [5]. These plans are part of the overall *Business Continuity Management (BCM)* strategy of an organisation. BCM is standardised by the *British Standards Institution (BSI)* and defined in [6] as *"a holistic management process that identifies potential threats to an organization and the impacts to business operations that those threats, if realized, might cause, and which provides a framework for building organizational resilience with the capability for an effective response that safeguards the interests of its key stakeholders, reputation, brand and value-creating activities."*

Part of the Business Continuity Management standard defined by the BSI is its lifecycle. Similar to the BPM lifecycle it consists of four phases that each serve an individual task [6]: (1) understanding the organisation, i.e. critical business processes, resources, and other entities, plus their respective potential threats are identified and, in case of their occurrence, potential consequences are predicted; (2) determining business continuity strategies, i.e. the minimum level of business operations to mitigate the business impact and specification of time frames until a normal operational level is restored; (3) development and implementation of BCM responses, e.g. accepting the risk, removing it, or installation of a *Recovery Plan*, that defines steps to be taken back to status quo ante in the given time frame; (4) exercising, maintaining, and reviewing, i.e. exposing flaws of the implemented BCM strategies and plans, which helps to improve them in the next iteration of the lifecycle. However, note that some threats are impossible to simulate in their full extent.

In the first step of the lifecycle the *Business Impact Analysis (BIA)* is addressed. According to [6], BIA is defined as the "...process of analyzing business functions and the effect that a business disruption might have upon them". A more specific description of BIA's purpose is given by [7]: it "...identifies, quantifies and qualifies the business impacts of a loss, interruption or disruption of business processes on an organisation and provides the data from which appropriate continuity strategies can be determined". One difficulty in BIA is to examine and extract resources and, with equal importance, the failure dependencies between them, i.e. if one resource gets unavailable, others do as well. The resulting failure dependency model represents the basis of further impact analyses.

3. STATE OF THE ART

To address the issue of Business Impact Analysis both domains have to be taken into account: business processes and resource infrastructures. However, in recent research they have been mostly regarded separately. Unfortunately, due to the fact that Business Continuity Management is not considered to be in the responsibility of the ICT department [6][13] not much computational support for business impact analysts is provided so far. Though, a couple of theories about resource dependencies in general can be found in exclusively business related literature, e.g. in [8], no modelling approach for their failure dependencies could be found by the author. Until today, business impact analysts still use tools like Microsoft Visio [21] to graphically model and understand the interconnections and possible impacts for an organisation in case a particular resource becomes unavailable.

However, BIA is about analysing the impact on the performance of business processes in the event of a disruption. A number of approaches for analysing the general performance of business processes as a part or as an extension of BPM suits already exist. Many of them, (e.g. [10]), are based on *Monolithic Model Transformation*. The principle of Monolithic Model Transformation is the direct transformation from a particular business process model and performance input parameters to an input for a specified performance analysis tool. Performance input parameters are in the form of historic, assumed, or planned data that is annotated to the business process and its elements to carry out a performance analysis. Examples of these parameters are: the process instance occurrence annotated to a process start element and the resource demand annotated to an activity.

The monolithic approach is restricted to only one single analysis tool and to one specific process model. However, currently networked business processes are composed of parts modelled and managed with different BPM environments. This motivates a more generic approach, namely *Decomposed Model Transformation*, that allows for an integration of multiple process modelling languages and different types of analysis tools. Figure 1 [11] shows its general concept. Note, that M2M stands for Model-to-Model transformation and M2T for Model-to-Text transformation.

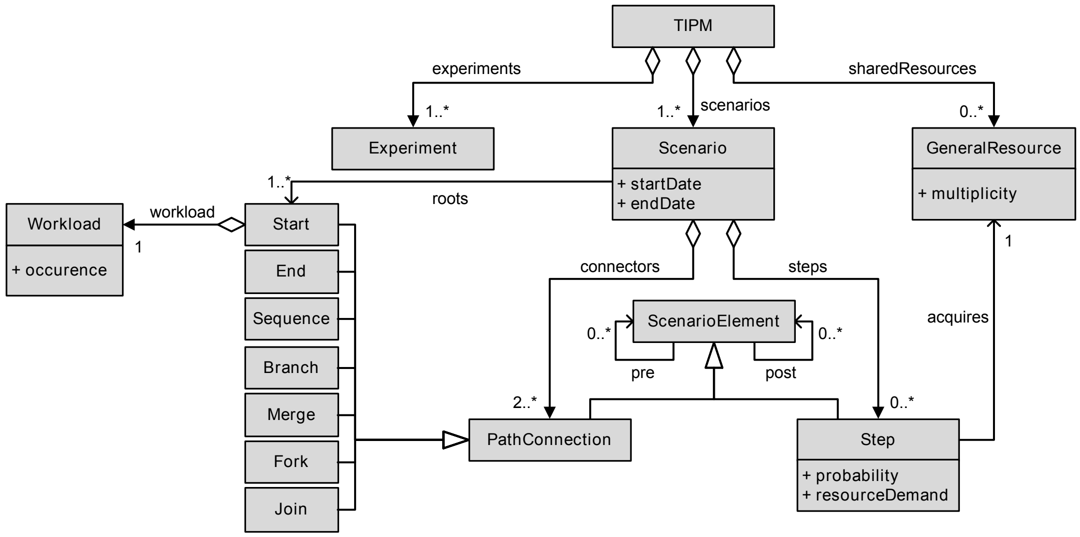

Figure 2: Tool Independent Performance Model (TIPM) Meta-Model

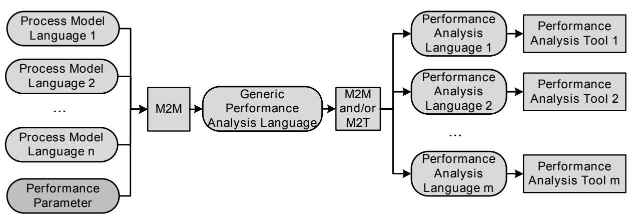

Figure 1: Integration concept via decomposed model transformation

The execution semantic of various different business process models can be abstracted into a unified Petri-Net like behaviour model. This similarity is utilised by a generically usable analysis model depicted as generic performance analysis model in the concept figure. In addition to expressing the process behaviour of a system defined by the input process models, the intermediate model also has to contain the performance parameters necessary for the different types of analyses. In [11], the language *Kernel LAnguage for PErformance and Reliabilty analysis (KLAPER)* is proposed as such a generic performance analysis model language. Other examples are the *Core Scenario Model (CSM)*, introduced in [12], and the *Tool Independent Performance Model (TIPM)* which has derived from CSM [9] and addressed some of its limitations, e.g. static parameter definition replaced by a more generic parameter concept. TIPM, of which a simplified version is depicted in Figure 2, comprises business process behaviour, performance data, and monitors to be filled with the results that are to be computed out of simulations and analytics.

It represents the generic performance analysis language in the Model-Driven Performance Engineering (MDPE) workbench to which every integrated business process modelling language has to be translated and from which a transformation to any analysis model of choice has to be performed.

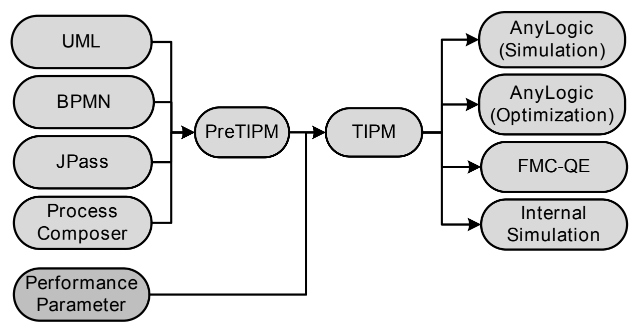

Figure 3: Model transformations in the MDPE workbench

The MDPE workbench extends existing process modelling tools, for example BPM suits, with performance related decision support [9]. To enable this extension it applies various MDE operations, e.g. decomposed model transformations, model annotations, model weaving, and megamodelling.

The actually implemented model flow of the MDPE workbench from source models to analysis models is depicted in Figure 3, the arrows representing model transformations. As shown, so far adapters for the business process modelling languages BPMN [14] and UML activity diagrams [15], as well as for the tools Process Composer of the Netweaver BPM process environment [16] and JCOM1 jPass [17] exist. The depicted *PreTIPM* is used as an intermediate model to enable the combination of several source models, i.e. models of different languages can yet be interconnected and as a result still be analysed by the MDPE workbench. As such, the PreTIPM model contains all the behavioural business process information extracted from the respective source models but already conforming to the TIPM language.

After merging several distinct process models into one, the integration of the performance parameters into the model is

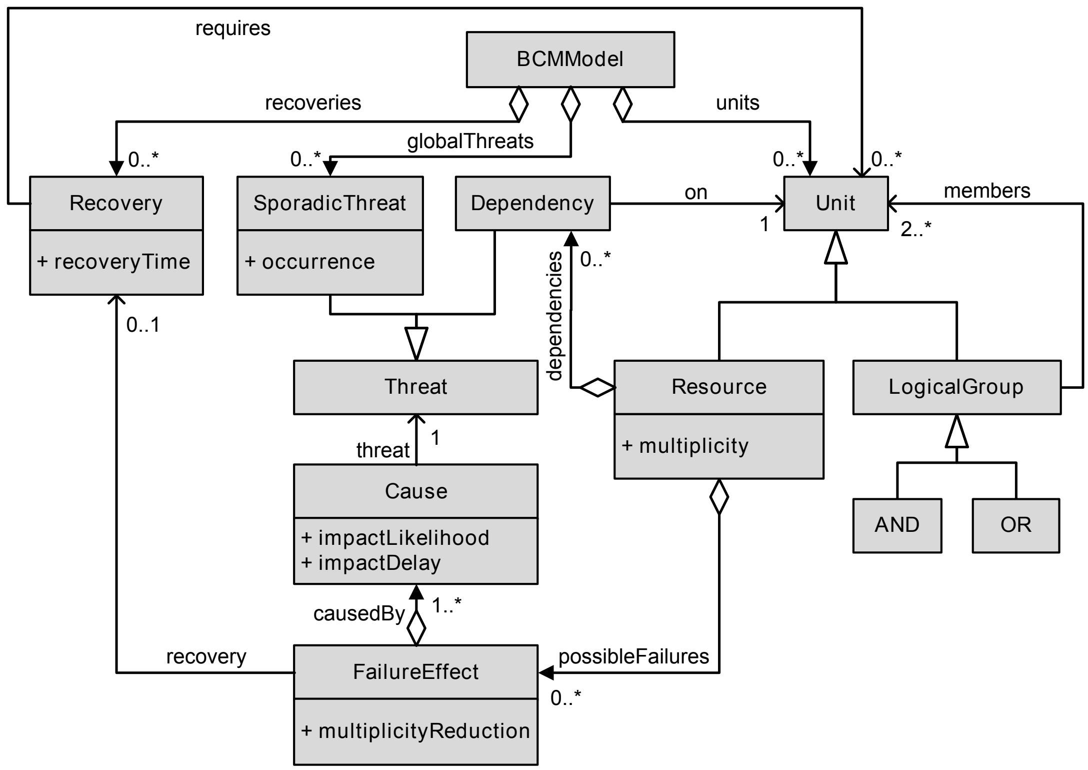

Figure 4: Generic BCM meta-model

performed. The result is then the TIPM model containing the business process data and the respective performance related data. In the last step of the transformation chain the TIPM is translated into the tool-specific performance model of choice. So far, adapters for the AnyLogic tool, suitable for performance simulations and optimisations [18], for the *Fundamental Modeling Concepts for Quantitative Evaluation (FMC-QE)* framework [19], and for an internal simulation tool had been implemented.

4. MOTIVATION TEST WHEN IT WOULD ACTUALLY BREAK

The MDPE workbench has been developed to address performance decision support for business processes. This explicit domain restriction of the tool for business processes leads to fairly limited modelling possibilities for resources. In order to respond to the additional requirements of the resource-centric BIA aspects, the workbench's lack of resource failure dependency and threat modelling abilities has to be addressed. The first identified limitation is the expressiveness restriction of the TIPM model on business processes and their directly related performance data only. In particular, two essential model limitations were identified:

- The first restriction of TIPM is that resources are independent entities only defined by their multiplicity. However, it has been pointed out in Section 2 that resources in practice are usually dependent on each other. Furthermore, according to the author's knowledge no modelling tool for this purpose is available.
- Another limitation is that no modelling possibility for threats that can cause disruptions in your resource infrastructure is provided.

5. INTEGRATED PERFORMANCE DECISION SUPPORT FOR PROCESS-CENTRIC BUSINESS IMPACT ANALYSIS

First, the model limitations had to be addressed to provide "Integrated Performance Decision Support for Process-centric Business Impact Analysis". Due to the unavailability of failure dependency modelling tools for resources a new language called *Generic Business Continuity Model (GenericBCM)* has been developed and was integrated into the model flow and workbench architecture. The GenericBCM meta-model is depicted in Figure 4. It represents the domain language for modelling resource dependencies in terms of disruption propagation. As such, it has to be able to express the following aspects:

1. The operating ability of a resources can be dependent on the operating ability of other resources. This *operating dependency* has to include also rather complex conditions, like "resource A becomes unavailable if either resource B or resource C together with resource D

is unavailable". The opposite of an operating dependency relation is called *failure dependency*, i.e. an occurring failure is delegated to the dependent resources.

2. The operating ability can not only be limited by other resources. An external event can occur which can force the resource to reduce or completely neutralise its ability to operate. Examples of these *threats* on resources are humans, which become sick, a computer that crashes, or a complete office that has to shut down due to a natural disaster.

3. Furthermore, it is essential to model the loss of the resource's operability. It is a difference if, e.g. a computer gets broken, which would correspond to a multiplicity reduction by one, or the power supply for the whole office breaks down, which would result in an overall computer multiplicity of zero.

4. After a resource suffers an operational mitigation, either caused by a failure dependency or by an external threat, there is the potential to recover after a certain amount of time. Examples are the recovery of a staff member after sickness or a computer getting repaired after a breakdown. The recovery of a resource can be a process itself.

The integration of resource dependency information provided by GenericBCM models entails an adaptation of the decomposed transformation chain of the MDPE approach (see Figure 3) towards an approach with multiple input domains. In particular, the transformation chain of the MDBIA workbench now has to incorporate two distinct but variable input sources, the business process information and the resource dependency information. Figure 5 depicts the applied model flow of the workbench including some of the supported model languages.

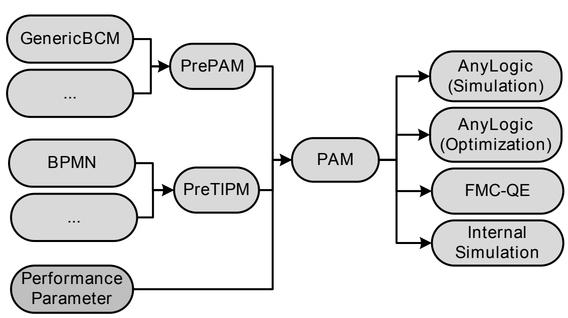

Figure 5: Model transformations in the MDBIA workbench

As a result of the merging of the business process and the resource failure dependency domains, the expressiveness of some of the already involved models has to be extended. The performance parameters have to be modified according to the increased modelling possibilities. In addition, the generic performance analysis model has to be exchanged as TIPM is only able to express business processes in combination with annotated performance information and analysis configuration data. In order to accommodate this request, the new model language *Performance Analysis Model (PAM)* (see Figure 6) was developed. It is based on the TIPM and extends it in a generic fashion to enable expressing information of resource dependency models, e.g. GenericBCM, without information loss. In PAM dependencies between resources are now modelled as corresponding resource behaviour nets, whose structure is mainly inspired by petri-nets and program flow charts. The result is a language in which business processes, threats and resources are the central elements. Business processes are modelled by commonly used units, such as activities and control flow elements. This has been adopted from the TIPM model. In contrast, resources and threats have attributes and behaviour nets containing of states and transitions. The behaviour nets have the purpose of altering these attributes and propagating impacts in the infrastructure during an analysis run.

The transformation chain acts as follows: At first the GenericBCM or any other integrated resource dependency model is translated into a PAM conform model called *PrePAM*. The resulting model contains all the information of the GenericBCM but in the more general notation of PAM. Parallel to that, the business process data from the business process models is translated into PreTIPM, just like in the MDPE workbench. Neither the PreTIPM, still conforming to the TIPM model language, nor the transformation towards the model had to be essentially modified. As the PAM extends the TIPM, any model conforming to TIPM is conforming to PAM, as well. Both results, the PreTIPM and the PrePAM are along with the Performance Parameters combined in the next transformation to the *PAM* model, conforming to the PAM model language and the sum of all information provided by these three input models. As the performance parameters contains links to both of the other input models, it is responsible for all the interconnections within the resulting PAM model, i.e. a resource demand defined in the performance parameter model possesses a link to the annotated activity in the process model and a link to the resource that is to be acquired in the dependency model.

The advantage of this generic approach is the reduction of number and complexity of the necessary transformations combining n business process model languages and m resource dependency languages with k performance analysis tools. Using a monolithic integration one would need $m*n*k$ rather complex transformations. The generic approach only needs $m+n$ transformations from the source model languages to the generic model, plus k transformations to the analysis tool models. As these kind of transformations are more functionally specialised their complexity is decreased in terms of lines of code.

In addition to the models, the transformations between them and analysis adapters need to be modified accordingly to enable BIA. Also, it has to be noted that each transformation additionally generates a tracing model, in order to enable an appropriate result management, i.e. results have to be traced back to their original source model elements.

The proposed concepts have been implemented in a workbench called *MDBIA*. It is based on the MDPE workbench[1] and reuses and extends its modelling concepts. In the workbench modelling is done utilising the capabilities of the Eclipse Modeling Framework (EMF) [24] and its extensions; transformations are implemented with the help of ATLAS Transformation Language (ATL) [25][26].

[1]The MDPE workbench is an eclipse application (see [23])

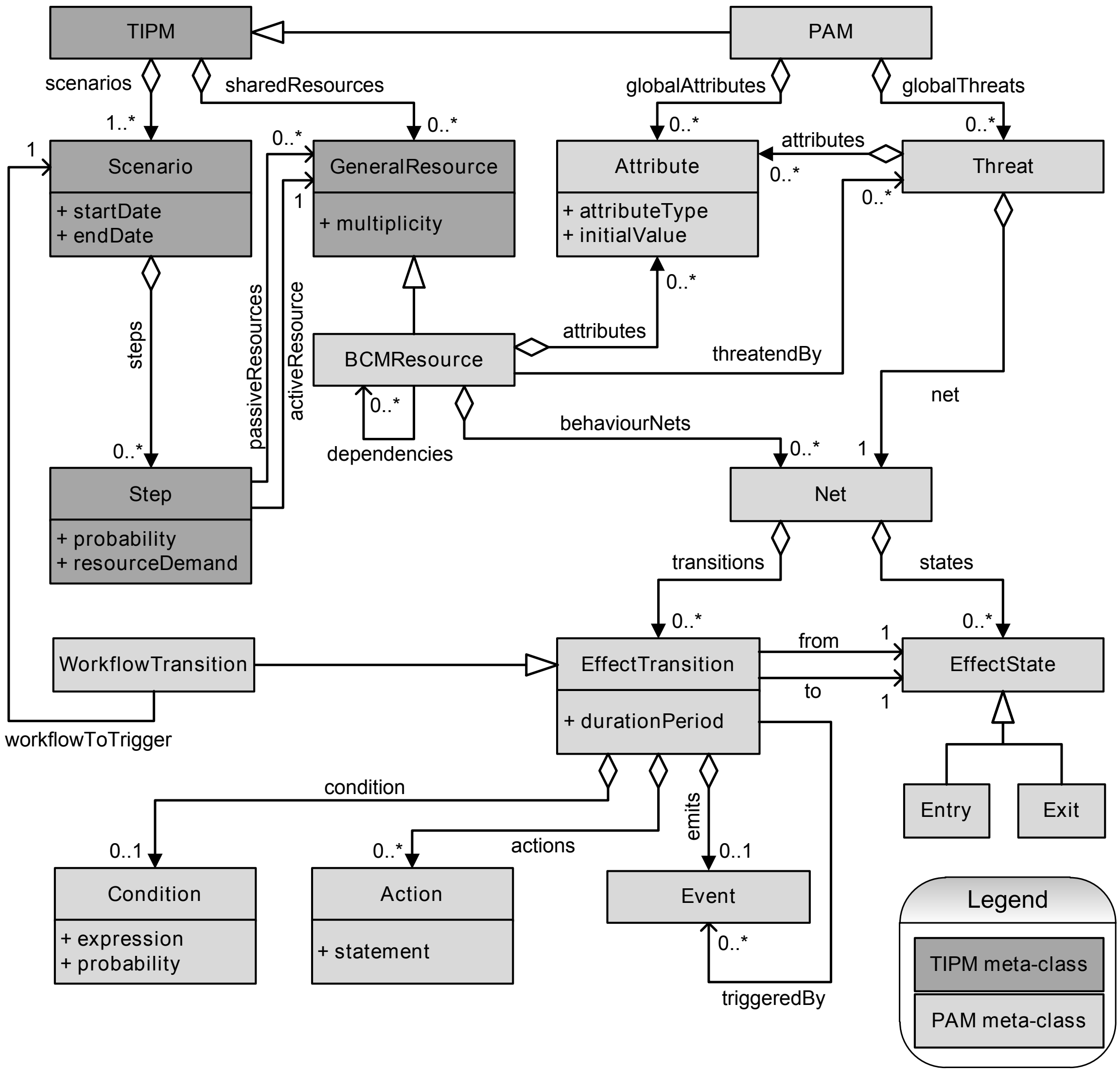

Figure 6: Performance Analysis Model (PAM) extension to TIPM model

6. EVALUATION

After having introduced the MDBIA workbench, its operability is evaluated in this section. This is done by examining a reasonable complex example and discuss the results, i.e. performing a qualitative analysis. First, the example's resource and threat model conforming to the GenericBCM meta-model, as well as a corresponding business process model and performance parameter model are introduced. Then in the intermediate PAM model is presented, which is a result of the transformations and represents all the information provided by the input models at once. Subsequently, the performance results of the analysis is shown and discussed. Thus, the final evaluation is performed by showing that the results are reasonable.

6.1 Input Models

The example case presented here is about the general topic of processing work packages. This use case study may seem simple at first but already produces results complex enough to perform a qualitative analysis of the concept. Note, all parameters correspond to the time unit *day*, e.g. a recovery time of 5 actually means five days.

Resource and Threat Model (GenericBCM).

Five resources are modelled, two actually carrying out work of the business process, "Technical Staff" and "Desktop PC", plus three indirectly involved resources, which are required by the others: "Power", "Office", and "UPS" (Uninterrupted Power Source). The example GenericBCM model is shown in Figure 7. In the use case the availability of the

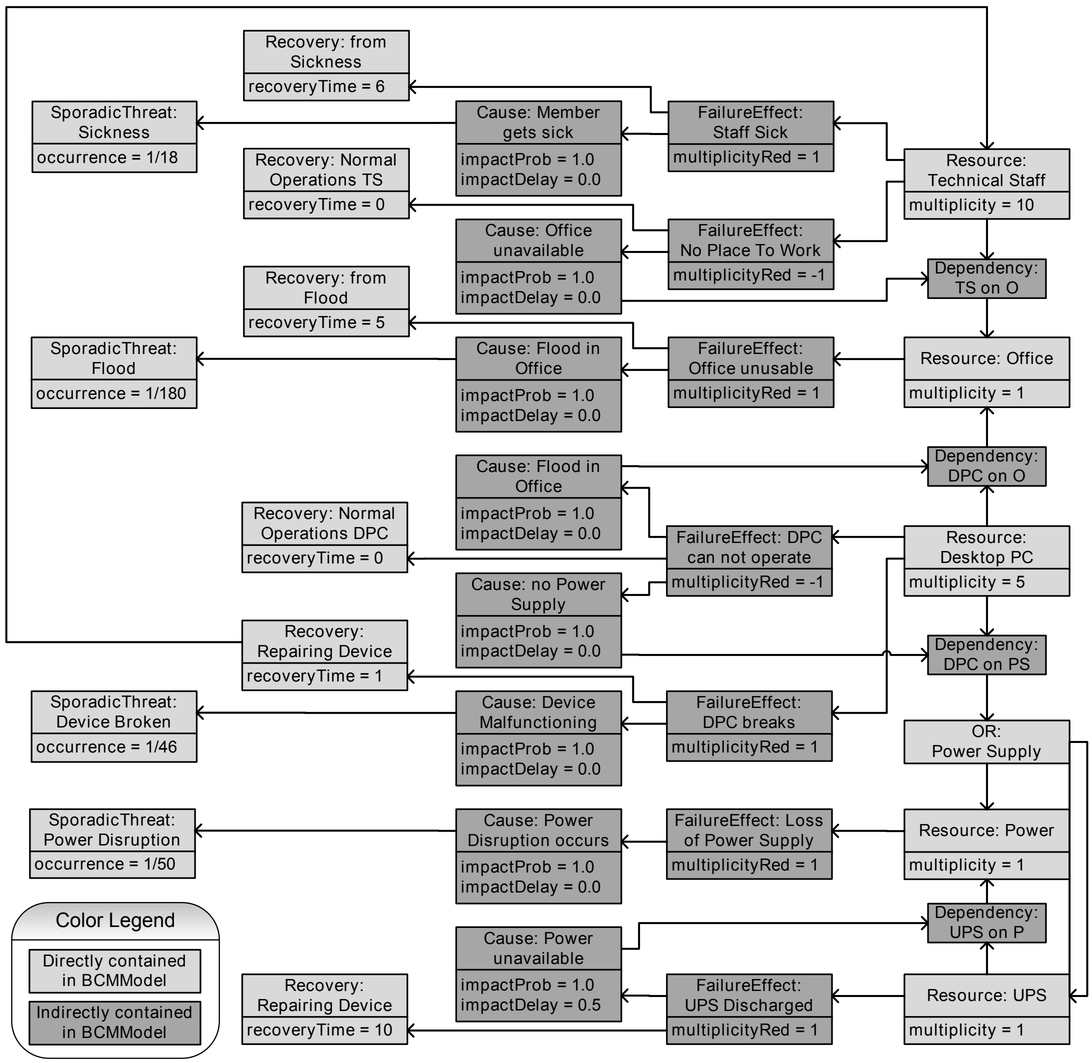

Figure 8: Complete GenericBCM model

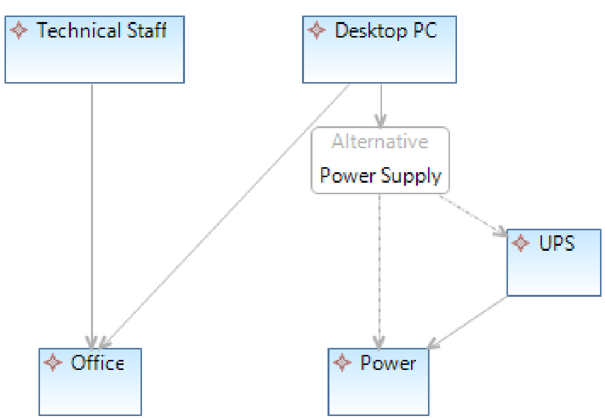

Figure 7: Simplified GenericBCM model in the editor

"Office" resource impacts the operability of both, "Technical Staff" and "Desktop PC". Furthermore, the latter one is dependent on "Power Supply" which is a LogicalGroup of the type OR and as such represents the availability of at least one of the resources "UPS" or "Power". "UPS" in turns is also dependent on "Power" as it has to be recharged after each usage.

However, many elements of the GenericBCM meta-model, such as Causes, Threats, and FailureEffects, are not displayed in the simple overview. The complete model is depicted in Figure 8. For clarity reasons, the BCMModel element is not displayed in the figure, but every element coloured in light grey is directly contained in it. The others are indirectly contained.

Notable features of the model sorted by resource are the following:

- Resource "Technical Staff" has a multiplicity of 10 and contains two FailureEffects that impact its operability: (1) "Staff sick", which is caused by the SporadicThreat "Sickness", occurring every 18 days, and reduces the multiplicity by 1; (2) “No Place To Work”, which is caused by “Office unavailable”, expressed by the Dependency “TS on O” (“Technical Staff” on “Office”), and reduces the multiplicity to 0, which is expressed by “−1”. The recovery from FailureEffect (2) happens instantly right after the “Office” is available again. In contrast, the Recovery “from Sickness” takes 6 days.

- Resource “Office” has a multiplicity of 1 and is threatened by “Flood”, which occurs every 180 days and causes the multiplicity to be reduced to 0. The Recovery “from Flood” takes 5 days.

- Resource “Power” has a multiplicity of 1 and is threatened by “Power Disruption”, which occurs every 50 days and causes the multiplicity to be reduced to 0. The Recovery “of Power” takes 1 day.

- Resource “UPS” (Uninterrupted Power Supply) is to provide power even though a “Power Disruption” occurred. This is expressed in the model through the FailureEffect “UPS discharged”, which reduces the Resource’s multiplicity from 1 to 0 after an impactDelay of 0.5 days. The impactDelay is modelled in the Cause “Power unavailable”, which further defines the relation between the FailureEffect “UPS discharged” and the Dependency “UPS on P” (“UPS” on “Power”). The recovery process of “UPS charging” takes then 10 days. This means. after “Power” becomes unavailable, the UPS is still up and running for half a day in order to provide power for the system and is recharged another 10 days later.

- Resource “Desktop PC” has a multiplicity of 5 and contains two FailureEffects that impact its operability: (1) “DPC breaks”, which is caused by the SporadicThreat “Device Broken”, occurring every 46 days, and reduces the multiplicity by 1; (2) “DPC can not operate”, which is caused by either “Office not usable”, expressed by the Dependency “DPC on O” (“Desktop PC” on “Office”), or “no Power Supply”, expressed by the Dependency “DPC on PS” (“Desktop PC” on “Power Supply”). The second FailureEffect consequently reduces the multiplicity to 0 if one of these Causes are triggered. As soon as these threats passed the recovery of this FailureEffect happens instantly. In contrast, the Recovery “Repairing Device” of FailureEffect (1) takes 1 day and, on top of it, needs a member of the “Technical Staff” to carry out the process.

Annotated Business Process Model.

The business process data of the example use case is available in the form of a BPMN model and displayed in Figure 9. It can be seen that the model consists of four activities, two of which are processed in parallel. Additionally to the plain process data, performance parameters have been annotated to the model. These are not shown in the figure but instead informally listed:

- The instance occurrence of the process is modelled by a normal distribution with the parameters 10.0/15.0/7.0 (most likely/worst case/best case) per day.

- The working time demand of the activity “Preparation” is modelled by a normal distribution with the parameters 0.1/0.2/0.05 days. The resources “Technical Staff” and “Desktop PC” are acquired in order to carry out this activity.

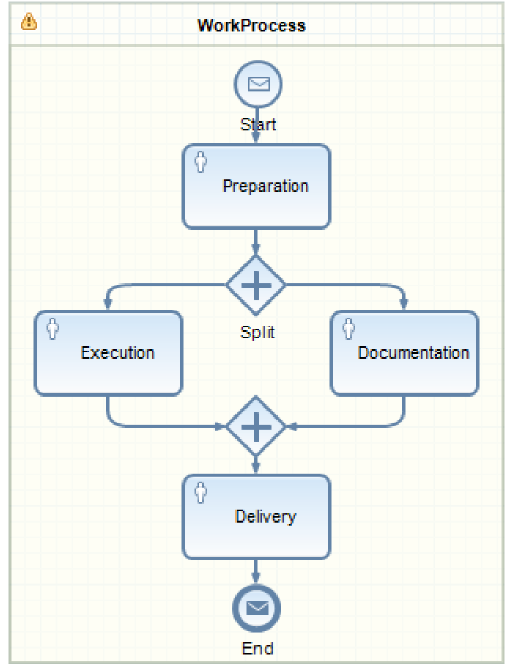

Figure 9: Example business process model conforming to the BPMN standard

- The working time demand of the activity “Execution” is modelled by a normal distribution with the parameters 0.4/0.5/0.25 days. The resource “Technical Staff” is acquired in order to carry out this activity.

- The working time demand of the activity “Documentation” is modelled by a normal distribution with the parameters 0.2/0.25/0.15 days. The resources “Technical Staff” and “Desktop PC” are acquired in order to carry out this activity.

- The working time demand of the activity “Delivery” is modelled by a normal distribution with the parameters 0.1/0.2/0.05 days. The resource “Technical Staff” is acquired in order to carry out this activity.

6.2 Intermediate Model

The previously described input models, namely, the threat and resource dependency model (GenericBCM), the business process model (BPMN), and the performance parameter model, are processed by applying the appropriate transformations described in Section 5. Before being handed over to the performance analysis adapter, in this case the internal simulation tool adapter, a generic intermediate model conforming to the PAM meta-model is built, of which a simplified version is shown in Figure 10.

Here, all the information provided by the input models is represented at once. The PAM model depicted comprises the original business process, now conforming to PAM meta-model, the recovery process “RepairingDevice”, and the five resources, introduced by the threat and resource dependency model, along with their behaviour nets. As each of the

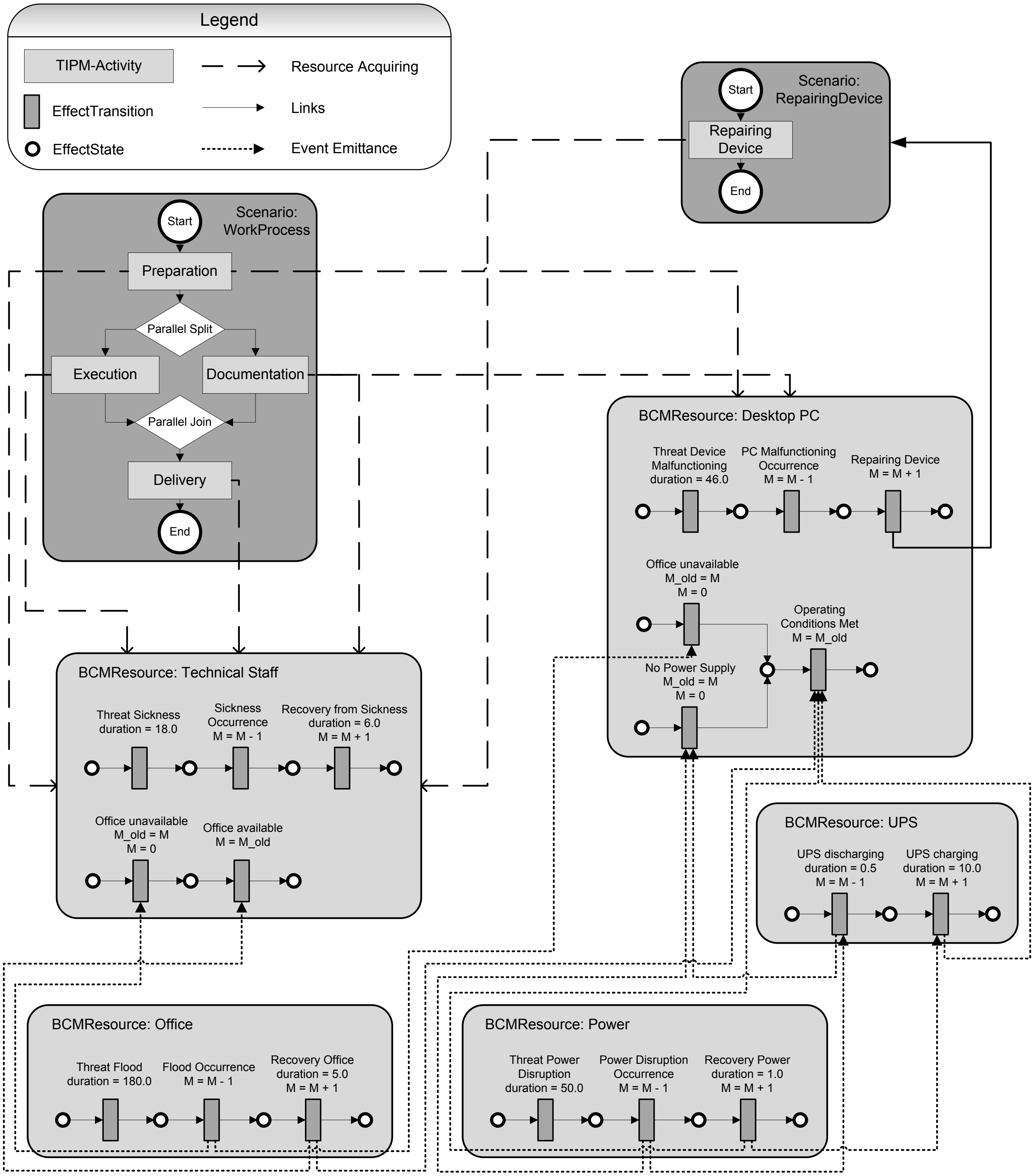

Figure 10: Simplified view on the PAM model resulting from the example input

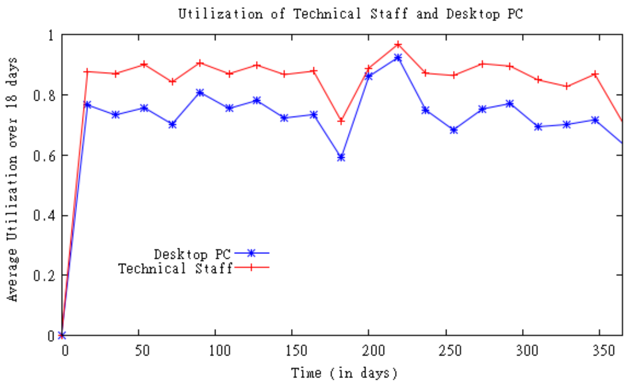

Figure 11: Utilisation of "Technical Staff" and "Desktop PC"

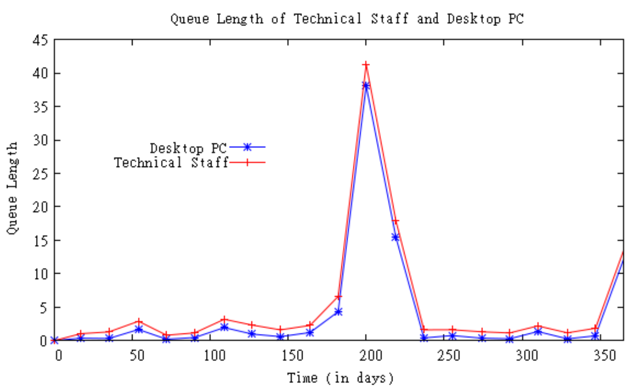

Figure 12: Queue Length of "Technical Staff" and "Desktop PC"

threats modelled were only impacting single resources, they have been included into the behaviour nets. Hence, no global threat can be found in the model.

Note that in order to improve the comprehension only essential parts of the PAM model are shown, so it is easier to grasp which relations between the individual elements exist. Examples for not depicted data are: some performance parameters, like working time demand and process instance occurrence.

6.3 Performance Results

The PAM model is the input for the internal simulation tool, which produces performance results. Selected performance output parameters are presented in this section, namely: the utilisation of the two resources "Technical Staff" and "Desktop PC" in Figure 11, the queue length of these resources in Figure 12, the gross working time of the process in Figure 13, and the process entries and process exits in Figure 14.

The time from 01/03/2013 to 01/03/2014 was simulated. The graphs are showing the results of the simulation.

6.4 Discussion

Taking a closer look at the results, a couple of effects become obvious. One of them is, that the simulation has a warm up phase of about twenty days, due to the initial idle

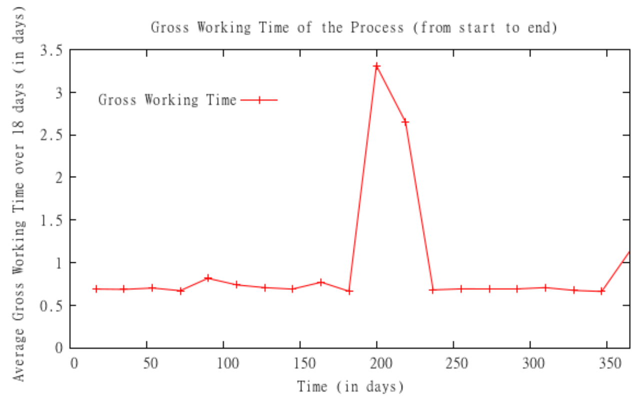

Figure 13: Gross Working Time of the Process

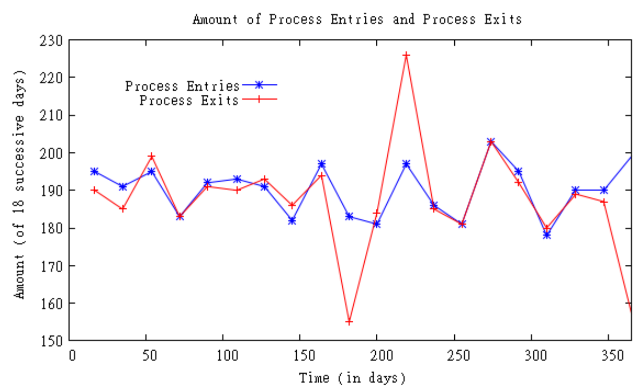

Figure 14: Comparing the numbers of process entries and process exits

state of the process and resources. After that short period, reasonable values can be extracted that are examined in the remainder of this section.

A significant impact on the performance results is caused by the occurrence of the "Flood" event, which influences both of the operating resources "Technical Staff" and "Desktop PC", at times 180.0 and 360.0. The impact of this threat on the resources's utilisations is a nosedive down to 0.6. In contrast, the corresponding queue length rises up to a number of almost 50, which is reasonable taking the forced unavailability over five successive days of both resources into account. The same effect is in some extent noticeable regarding the gross working time depicted in Figure 13: While the average time to complete the process from start to end is about 0.75 days, this value rises up to 3.5 in the period right after a "Flood" occurrence. With respect to the average number of process exits shown in Figure 14, it can be seen that a first reaction is a drop from 190.0 to 160.0 due to the process being blocked for five days. Right after that, the system reacts with a significant increase up to 225.0 as it has to additionally process the bottled-up workload in order to make up for the omissions of the five-day disruption. After approximately forty days, the system has fully recovered from the "Flood" incident.

Another, rather smaller, impact on the system can be recognised for the "Power Disruption" threat. Especially, in the queue length parameter a reaction is noticeable: after each period of 50 days, which corresponds to the occurrence rate of this threat, a slight increase of the resources's queue length can be identified. Because of the "UPS" is reducing the off-time of power supply to only half a day, the effect is almost imperceptible with respect to the other displayed result parameters.

Generally, the performance results for the resources "Technical Staff" and "Desktop PC" appear to be almost identical. The reason for that is identified in the first process activity "Preparation", which acquires both resources. Hence, if only one of the resources is not available, this activity cannot be processed and thus represents a bottle neck of the system. As the first activity is blocking the execution of the successive activities, the rest of the system stays mostly idle.

Some of the frequently reoccurring threats, like "Sickness" or "Device Malfunctioning" seem to have no effect on the system at all. In order to examine if they are actually happening, it is sufficient to have a look at specific events of the simulation event log. In the following box an excerpt of the log is shown, only including events representing changes of the resources's multiplicities. Thereby, "mult" is the new adapted multiplicity of the resource "res". Additionally, with "ql" the queue length of the resource at this point of time (first value) is given.

Excerpt of the Simulation Adapter Event Log

> 180.0; res: Office; act: MULTIPLICTY_CHANGE; mult: 0; ql: 0
> 180.0; res: TechnicalStaff; act: MULTIPLICTY_CHANGE; mult: 0; ql: 0
> 180.0; res: DesktopPC; act: MULTIPLICTY_CHANGE; mult: 0; ql: 0
> 185.0; res: Office; act: MULTIPLICTY_CHANGE; mult: 1; ql: 0
> 185.0; res: TechnicalStaff; act: MULTIPLICTY_CHANGE; mult: 10; ql: 63
> 185.0; res: DesktopPC; act: MULTIPLICTY_CHANGE; mult: 5; ql: 53
> 198.0; res: TechnicalStaff; act: MULTIPLICTY_CHANGE; mult: 9; ql: 26
> 200.0; res: Power; act: MULTIPLICTY_CHANGE; mult: 0; ql: 0
> 200.5; res: UPS; act: MULTIPLICTY_CHANGE; mult: 0; ql: 0
> 200.5; res: DesktopPC; act: MULTIPLICTY_CHANGE; mult: 0; ql: 24
> 201.0; res: Power; act: MULTIPLICTY_CHANGE; mult: 1; ql: 0
> 201.0; res: DesktopPC; act: MULTIPLICTY_CHANGE; mult: 5; ql: 33
> 204.0; res: TechnicalStaff; act: MULTIPLICTY_CHANGE; mult: 10; ql: 29

In the excerpt of the event log the behaviour of the system is demonstrated, when, for example, the "Office" becomes unusable due to an occurring "Flood" at time 180.0. Instantly, the two depending resources "Technical Staff" and "Desktop PC" become unavailable and recover five days later after the "Office" has recovered. Also, the multiplicity reduction due to "Sickness" at time 198.0 and the corresponding recovery at day 204 can be seen.

Another effect demonstrated by the event log excerpt is the occurrence of "Power Disruption" at time 200.0. Half a day later, also the "UPS" along with the "Desktop PC" is affected. Their multiplicity decreases to zero, which represents a complete loss of their operability. Right after "Power" recovered, the operability of "Desktop PC" is restored at time 201.0.

Considering the provided discussion in this section, the performance results have been determined to be reasonable. To conclude the evaluation, it is therefore suggested the PAM model as well as the simulation adapter are operating in a reasonable fashion.

7. CONCLUSION AND FUTURE WORK

This paper introduced a solution for integrated performance decision support for process-centric business impact analysis, namely the MDBIA workbench. Its main purpose is the prediction of involved consequences on the business process level, taking into account occurring disruptions in the resource infrastructure. This has been implemented by adopting concepts of the MDPE workbench along with its generic approach of analysing the performance of various business processes. For reasons highlighted in Section 4, these concepts do not fully meet the requirements of Business Impact Analysis. Thus, the identified limitations of the MDPE workbench have been addressed by extending it with modelling and analysis possibilities for complex resource infrastructures. The resulting workbench integrates a new modelling language called GenericBCM, which allows to model resources, operability dependencies and threats. Furthermore, a new generic intermediate model language, called PAM, has been designed with the purpose of addressing the additional requirements of Business Impact Analysis.

The MDBIA workbench is used in the context of real world industrial use-cases provided by the EU-funded project TIMBUS, namely in the areas of dam safety and eHealth. However, when applying the method presented in this paper to the use cases a few limitations and possible improvements became apparent. This is why we propose the following modifications and extensions to be future work:

1. In the current GenericBCM model parameters are expressed as static values. To increase model accuracy and model expressiveness it is suggested to introduce more advanced parameter representations, e.g. distribution functions, to achieve improved results.

2. Although, there is currently no other language than GenericBCM for the explicit purpose of modelling resource failure dependencies, a number of resource landscape modelling possibilities are available, e.g. the Topology Editor of IBM RSA [20]. From these models the dependencies could potentially be imported.

3. The intermediate PAM model is domain-specific. But with regards to PAM's behaviour net semantic it is to some extent already close to a General Purpose Language (GPL). One promising modification would be to replace this model language with a well researched GPL, for example Coloured-Petri-Nets [22], for which a number of analysis tools can be readily be deployed.

4. The internal simulation tool uses a single-thread execution model to avoid concurrent access to the PAM

model. A future implementation could provide a multi-threaded execution of the internal simulation tool. Another alternative to speed up the current analysis is to replace the simulation, which can be slow for large models, with a more direct and faster analytical approach.

5. The MDBIA workbench is used as a design-time tool for BIA. A possible extension would be to provide support for real-time monitoring, analysis and disaster management. Therefore real-time events could be captured and consumed for continuous BIAs. This approach would enable short-term decision support in real-time.

8. REFERENCES

[1] Library of Congress. Congressional Research Service: The Economic Effects of 9/11: A Retrospective Assessment. http://www.fas.org/irp/crs/RL31617.pdf (2002)

[2] van der Aalst, W., ter Hofstede, A., Weske, M.: Business Process Management, International Conference, BPM 2003, Eindhoven, The Netherlands, June 26-27, 2003, Proceedings, volume 2678 of Lecture Notes in Computer Science. Springer (2003)

[3] Fritzsche, M., Picht, M., Gilani, W., Spence, I., Brown, J., Kilpatrick, P.: Extending BPM Environments of Your Choice with Performance Related Decision Support. In: Proceedings of the 7th International Conference on Business Process Management, pp. 97–112. Springer-Verlag, Berlin, Heidelberg (2009)

[4] Harris, S.:CISSP All-in-One Exam Guide, Fifth Edition. McGraw-Hill, Inc., New York, NY, USA (2010)

[5] Janchivlamdan, R. Buchanan, T., Tierney, T., Davis, J.: Business Continuity Planning and Development. https://ectd.du.edu/source/uploads/11142668.pdf (2007)

[6] British Standards Institute. BS25999, Code of Practice for Business Continuity Management. BSI (2006)

[7] Smith, D.: Business Continuity Management: Good Practice Guidelines, pp. 0–18. BSI (2002)

[8] Pfeffer, J., Salancik, G.: The External Control of Organizations: A Resource Dependence Perspective. Stanford University Press, Stanford, California (2003)

[9] Fritzsche, M.: PhD Thesis: Performance related Decision Support for Process Modelling. School of Electronics, Electrical Engineering and Computer Science, Queens University Belfast (2010)

[10] Rozinat, A., Wynn, M., van der Aalst, W., Hofstede, A., Fidge, C.: Workflow Simulation for Operational Decision Support Using Design, Historic and State Information. In: Proceedings of the 6th International Conference on Business Process Management (BPM'08). Springer-Verlag (2008)

[11] Grassi, V., Mirandola, R., Sabetta, A.: From design to analysis models: a kernel language for performance and reliability analysis of component-based systems. In: Proceedings of the 5th international workshop on Software and performance, pp. 25–36. ACM, New York (2005)

[12] Petriu, D., Woodside, M.:An intermediate metamodel with scenarios and resources for generating performance models from UML designs. In: Software and Systems Modeling, Volume 6, Issue - 2, pp. 163–184 (2007)

[13] Stranack, T., Cornish, C.: Business Continuity Management - Bridging the divide. In: Business, vol. 6, pp. 1–4 (2009)

[14] Object Management Group. Business Process Modeling Notation Specification, Final Adopted Specification, Version 1.0. http://www.bpmn.org/Documents/BPMN_1-0.pdf (2006)

[15] Object Management Group Inc: Unified Modeling Language 2.0: Superstructure. http://www.omg.org/spec/UML/2.0/Superstructure/PDF. formal/05-07-04, 2005

[16] Snabe, J., Rosenber, A., Molle, C., Scavillo, M.: Business Process Management: The SAP Roadmap (2008)

[17] Metasonic AG. jPASS! - Subjektorientierte Prozessmodellierung. http://www.metasonic.de/ (2009)

[18] XJ Technologies. AnyLogic — multi-paradigm simulation software. http://www.xjtek.com/anylogic/ (2010)

[19] Porzucek, T., Kluth, S., Copaciu, F., Zorn, W.: Modeling and Evaluation Framework for FMC-QE. In: Proceedings of the 2009 16th Annual IEEE International Conference and Workshop on the Engineering of Computer Based Systems, pp. 237–243. IEEE Computer Society, Washington DC, USA (2009)

[20] IBM. Rational Software Architect.http://www-01.ibm.com/software/awdtools/swarchitect/ (2011)

[21] Microsoft, Office Visio, http://office.microsoft.com/en-us/visio/ (2007)

[22] Jensen, K., Kristensen, L. M.: Coloured Petri Nets - Modelling and Validation of Concurrent Systems, Springer (2009)

[23] Eclipse Foundation: The Eclipse Foundation. http://www.eclipse.org/org/ (2013)

[24] Budinsky, F., Brodsky S. A., Merks, E.: Eclipse Modeling Framework. Pearson Education (2003)

[25] Jouault, F., Allilaire, F., Bézivin, J., Kurtev, I.: ATL: A model transformation tool. In journal Science of Computer Programming, vol. 72, pp. 31–39, Elsevier North-Holland (2008)

[26] Allilaire, Bézivin, J., F., Jouault, F., Kurtev, I.: ATL: Eclipse Support for Model Transformation. In: Proc. of the Eclipse Technology eXchange Workshop (eTX) at ECOOP, (2006)

Project partially funded by the European Commission under the 7th Framework Programme for research and technological development and demonstration activities under grant agreement 269940, TIMBUS project (http://timbusproject.net/).