Visual Workflow Process Modeling and Simulation Approach Based on Non-Functional Properties of Resources

Kawtar Ougaabal, Grégory Zacharewicz, Yves Ducq, Saïd Tazi

▶ **To cite this version:**

Visual Workflow Process Modeling and Simulation Approach Based on Non-Functional Properties of Resources

Kawtar Ougaabal [1], Gregory Zacharewicz [2],[*] , Yves Ducq [1] and Said Tazi [3]

[1]   Univ Bordeaux—IMS UMR CNRS 5218, 33405 Talence CEDEX, France;
      kawtar.ougaabal@u-bordeaux.fr (K.O.); yves.ducq@u-bordeaux.fr (Y.D.)
[2]   IMT Mines Ales—LG2IP, 30319 Alès CEDEX, France
[3]   University of Pau et Pays de l'Adour—LAOSI, 64600 Anglet, France; said.tazi@univ-pau.fr
[*]   Correspondence: gregory.zacharewicz@mines-ales.fr; Tel.: +33-466-785-000

check for updates

Featured Application: The methodology proposed in this paper may benefit industrials to better bridge the gap between business requirements and implementation requirements by using a tool merging modeling and simulation in the same phase. The modeling and simulation techniques proposed are based on a resource-perspective approach.

Abstract: With the emergence of big data and cloud technologies, companies now evolve in complex IT environments. This situation requires good practices for data process automation to be adopted to ensure system interoperability. Visual computing helps companies to describe and organize the ways in which information systems, devices, and people must interact. It incorporates a number of fields including modeling and simulation (M&S). This paper focused on M&S of data workflow processes that are key steps to bridging the gap between business views and goals on the one side, and operational implementations on the other side. Simulation adds a dynamic view to static modeling; it increases understanding of the behavior of process mechanisms and the identification of weak areas in data flow. Several research projects have been focused on control flow and data flow, but less attention has been paid to resource characteristics. This work is based on the MDSEA approach and the eBPMN language, and proposes an approach that aims to distinguish the types of resources carrying out process tasks. Furthermore, it introduces a new composite resource made from the relationship between a user (human resource) and a task form (IT resource). Moreover, it proposes a resource aggregation based on process performance combination in order to run and display a global performance measurement of a process path.

Keywords: workflow; data process; visual modeling & simulation; resources; BPMN; MDSEA

1. Introduction

Nowadays, the changing economic environment is characterized by a diverse and constantly evolving market, where responsiveness, flexibility, and innovation are the new criteria for success. In order to meet this need, business markets are looking for new manufacturing technologies to customize their products/services according to their customers' needs. Industry 4.0 is one of the concepts leading organizations to become more efficient, less wasteful, and much more productive. Industry 4.0 is an approach that aims to create an integrated and connected ecosystem to gain optimal results. It is mainly aimed at the manufacturing field, but can be applied to any other type of organization as shown in Reference [1].

In order to avail of the benefits of this paradigm, companies need to master their activities. To achieve this, they tend to use process automation, since it is one of the most effective methods for reaching this goal, according to Reference [2]. Several researchers have proposed methodologies for process automation. The initial goal was, and still is, to enhance and improve process modeling techniques by proposing tools and modeling languages to simplify the process and make it more accessible to all as shown in Reference [2]. The aim is to bridge the gap between business requirements and the technology used. That is why, over the last decade, «model-driven» engineering (MDE) approaches have emerged, considering models not just as documentation artifacts for communication purposes, but as central elements of the software engineering process in order to create a link between business requirements and IT requirements, as explained in Reference [3]. The methodology consists of smoothly describing a business need from a high level of abstraction to the concrete solution through model transformation. Even so, models provide solely a static presentation of the need. That is where the importance of visual computing comes in.

Visual computing deals with everything related to computer visuals. It is a discipline that covers the visual representation of data through various techniques. It encompasses a number of fields including visual analytics, augmented reality, visual image processing, modeling, and simulation according to Reference [4]. Today, visual computing is considered a key enabling technology for Industry 4.0 and the industrial internet as explained in Reference [5]. This paper focused on the modeling and simulation (M&S) process. Once a model is established, simulation aims to add a dynamic component to the static model in order to increase its understandability and clarity, or to discover unknown issues, as explained in Reference [6]. Verification in traditional process automation usually occurs post-implementation, which leads to errors and inaccuracies in the expected results. By this stage, the off-target requirements are much more difficult and costly to fix. By detecting them during the earlier "designing" phase, simulation helps to reduce design errors and allow incorrect requirements to be fixed before they occur [6]. As such, the simulation tool is essential for evaluating the performance of the process in terms of its behavior and dynamicity. The main purpose of the simulation approach is to be able to bring to business users an early evaluation of the future process they want to set up. It aims to experimentally consider different solutions and to compare them in order to support decision-making. In this way, various case-study simulations could be performed before their implementation. Of course, experimenting with several alternatives within a real-life system is not feasible as it would be time- and money-consuming, risky (if the solution fails), and is generally not a proactive approach.

Despite the wide availability of simulation methods, their application in the business processes field still remains a challenge, owing mainly to the fact that business analysts are searching for the easiest approaches by which to model and simulate their processes at the same time. The proposed approaches in the literature require some expertise in simulation and in the field of IT. Moreover, the conceptual model can be different than the simulated model and many tests are required to ensure their compatibility, which is money- and time-consuming. There are user-friendly tools on the market to overcome these limitations, but they are less adaptable and often very inflexible in their target outcomes, as mentioned in Reference [7]. In addition to that, according to Reference [8], several business-process modeling and simulation works focused on control flow and data flow, and less attention has been dedicated to the resources used, while resources are the inputs for producing goods and services. Thus, merging process modeling with simulation resources helps support accurate decision-making before implementing any solution.

To successfully improve their business information systems and process automation, companies call on computer engineering and services societies (IT companies) to better help their clients to identify process automation barriers and other opportunities to overcome productivity gaps. IT companies require a tool that merges process modeling and process simulation simultaneously. In order to reach the goals set by the client, the modeling simulation tool methodology should be understandable by all

stakeholders, from the business analyst to the IT consultant, in order to increase communication and facilitate the analysis as described in Reference [9].

For this, a tool based on an MDE approach is required. Such a tool has the ability to separate the operational needs from the technical needs, whilst still being able to highlight any links between the two. In addition, it couples the methodology with a modeling language, allowing different levels of precision to be modeled, first capturing high-level activities and then concentrating down to lower levels. Finally, the same modeling language includes the representation of non-functional requirements with simulation means to assess them.

For this purpose, this paper proposes an MDE analysis methodology with a tool merging modeling and simulation in the same phase, using a resource-perspective approach. Our work was based on «model-driven» system engineering architecture (MDSEA) [10] and the extended Business Process Modeling and Notation (eBPMN) language [7].

The originality of this contribution is in distinguishing resource types at the modeling and simulation phase (human, physical means, and IT resources). Moreover, this paper introduces a new composite resource simulation, considering the relationship between a user (human resource) and a task form (IT resource). In order to compute and display a global process measurement, a resource aggregation approach is proposed. Finally, the resulting tool displays the non-functional simulation results, going further than previous works by giving path information instead of task-only information.

The rest of the paper is structured as follows: Section 2 provides the materials and methods used to produce the proposed paper. Section 3 presents the contribution to overcome the above-mentioned issues. Finally, a real case study is presented to validate the approach, followed by a discussion in Section 4.

2. Materials and Methods

2.1. Model Driven Service Engineering Architecture

«Model-driven» engineering (MDE) is a software engineering methodology that considers models to be the core asset of the software development process as explained in Reference [11]. «Model-driven» architecture (MDA) [12] and «model-driven» interoperability (MDI) [13] are MDE initiatives. They aim to tackle, respectively, architecture and interoperability problems at earlier stages. From the same perspective, researchers proposed a unified methodology, «model-driven» service engineering architecture (MDSEA), based on MDA/MDI and designed to overcome their lack of modeling and developing services [10]. One of the main benefits of MDSEA is the definition of the three types of service system components (IT, Organization/Human, and Physical Means) and their integration at the highest level of the modeling phase. Therefore, in comparison to other «model-driven» approaches (MDA/MDI), the goal of MDSEA is not only IT-oriented implementation, but all types of resource implementation.

As shown in Figure 1, MDSEA is a three-layer methodology:

- Business system model (BSM): This top layer gives a general model of the system (top BSM). Leading down from this (bottom BSM) is the breakdown of the three resource domains, giving the details for each individually;
- «Technology-independent» model (TIM): This level focuses on describing the operational details of the system, including the three components, without detailing any implementation specificities;
- «Technology-specific» model (TSM): This level focuses on describing the specificities of each resource implementation.

A detailed explanation of each level of MDSEA is given in Reference [10].

There are different dimensions that can be addressed when using enterprise modeling to model a system: Processes, Products/Services, Organization, Information, and Environment, according to Reference [14]. This paper focuses on enterprise processes modeling only.
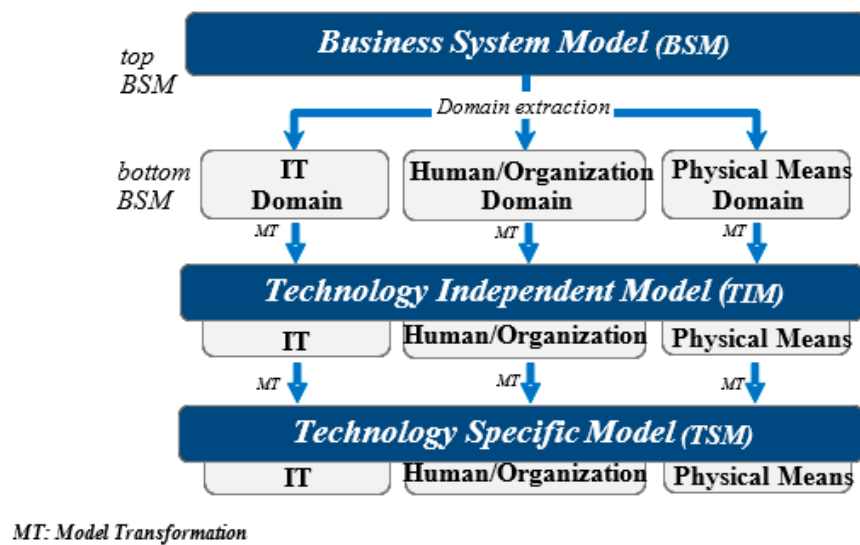
Figure 1. MDSEA methodology.

2.2. Business Process Modeling Notation

Several business process modeling languages are used in practice to describe processes from a general perspective and to establish their operational requirements, such as unified modeling language (UML) activity diagram, event-driven process chain (EPC), and business process modeling and notation (BPMN). BPMN is considered the de facto standard for process modeling. It is a graphical notation for modeling business processes that provides a standard notation in a way that is widely understandable to different users involved in process automation. It is both easy to learn and powerful enough to depict potential complexities of business processes. BPMN 2.0 was selected as the modeling language used in this paper's proposition for the following reasons [15]: (i) It aims to provide an notation that is understandable to both business and technical users; (ii) it guarantees the portability of the model, meaning that users could take a BPMN model created in one vendor environment and use it in another one; (iii) it is associated with a set of execution languages like business process execution language (BPEL); and (iv) it provides an extension mechanism that allows the introduction of new elements to the language, keeping its specificities unaffected.

2.3. eBPMN

Several simulation approaches and techniques have been provided in the literature. However, the effective use of business process simulation is still limited in practice for many reasons, as addressed in References [7,16]. To overcome these limitations, a couple of studies have proposed extensions of BPMN with non-functional requirements. Two types of solution have been proposed; the first type proposed a BPMN extension by adding new elements to its metamodel-like table format, new decorators as described in References [17–19] which alter the specificities of BPMN. The second type proposed the use of text annotation to represent additional needs, thus respecting the extension instructions provided in the BPMN specifications [20,21]. This paper describes an enhancement of the second type of proposed solution to maintain the benefits of BPMN.

Among the proposed tools, this work enhanced the eBPMN language [7]. Bocciarelli et al. (2011) [20] introduced PyBPMN, a lightweight extension of BPMN that includes performance and reliability measurements using text annotations. Moreover, they proposed an approach merging the modeling and simulation phases at the time of design by introducing their eBPMN language. eBPMN

is a language based on a «model-driven» method (model-to-model and model-to-text transformations) that automatically builds executable simulation code from a BPMN model [7]. The «model-driven» business process simulation method starts with a BPMN model that includes both functional and non-functional measurements using text annotation. The model-to-model (M2M) transformation takes as input the annotated BPMN model and yields as output the corresponding PyBPMN model. Then, the model to text (M2T) transformation, takes as input the PyBPMN model serialized into XMI and yields as output the corresponding eBPMN code. The M2M transformation is implemented by the use of OMG's language; QVT-O and M2T transformation is implemented by the use of Acceleo/OCL. More details are described in Reference [7].

The advantages of this approach are to:

- Merge modeling and simulation in the same step using a «model-driven» method to automatically build executable simulation code from BPMN;
- Guarantee the compatibility of the model by not modifying the BPMN metamodel and implementing eBPMN according to the BPMN execution semantics using token concepts;
- Allow non-functional simulation through process resources by using text annotation;
- Take into consideration composite resources;
- Handle non-predictable variables by using pseudo-random number generation and exponential distribution approximation.

This work improved the eBPMN resource approach [22] by distinguishing a resource's type as defined by MDSEA methodology in the design and simulation phases, as each resource has its own characteristics, requirements, and constraints in a business process.

2.4. Performance Aggregation

The goal of performance aggregation is to combine the performance measurement of each component of a system to obtain its total performance measurement. The configuration of the tasks could be in sequential, parallel, or alternative structures.

In order to solve interoperability issues, Heguy et al. (2019) [23] combined the stochastic workflow reduction (SWR) algorithm [24] with the work of Reference [25] to aggregate the performance measurements for the tasks shaping a process.

SWR computes the overall workflow performance by applying a set of reduction rules iteratively to the process, until one global activity is reached. The works described in Reference [25] determined the different aggregation typologies by using the decomposition technique. This paper focused on time and reliability indicators, as shown in Table 1. Following the example of task performance aggregation, this work applied the combination of these two methods to resource performance aggregation.

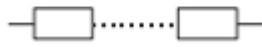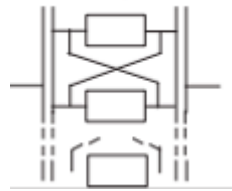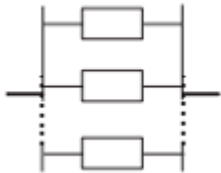**Table 1.** Performance aggregation computation.

Configuration	Time	Reliability
SEQUENTIAL	$\sum_{i=1}^{t} Ti$	$\prod_{i=1}^{t} Ri$
AND	$Max(T1,\ldots,Ti)$	$\prod_{i=1}^{t} Ri$

Table 1. *Cont.*

Configuration	Time	Reliability
	$Max(T1,\ldots,Ti)$	$Min(R1,\ldots,Ri)$

3. Results

3.1. Methodology Description

The methodology proposed in this paper is based on the MDSEA approach as described in Section 2.1, using the BPMN and eBPMN languages as described in Sections 2.2 and 2.3. The methodology can be applied by IT companies characterized by a diverse client portfolio with different goals to reach. The business analyst drives the proposed method, since they are responsible for business analysis and the gathering of requirements from several stakeholders relating to the system to be improved and automated. Each step of the methodology is described below, including the personnel involved (Figure 2):

- **TOP BSM:** This step aims to clarify conceptually with the client functional and non-functional requirements from the specification needs to provide a better understanding of the goals to be reached. This is followed by the identification of resource requirements to accomplish these objectives using grid evaluation, as explained in Section 3.2.1. The business analyst, as well as any other analysts (IT and other sectors) involved in the study system are included, as gathering information and reaching agreements are collaborative tasks.
- **BOTTOM BSM:** This step consists of modeling more formally the process at a high level of abstraction using the BPMN language. For this step, resource distinction is included by using only the task types provided by BPMN (user task, manual task, service task, etc.). This step does not require the presence of an IT analyst, since the technical implementation details are not discussed.
- **TOP TIM:** This step models detailed process operations using BPMN by describing operational details whilst hiding ones of any particular implementation. A path verification is then carried out by explicitly choosing one to check, or by assigning 50% probability to each process path to display them all. This mechanism is described in Section 3.5. If the desired paths are reachable and are compliant with the functional requirements, then the next step can be carried out. As with the BOTTOM BSM step, an IT analyst's presence is not required. Before moving on to the next phase, it is important to verify that all the desired paths to be studied are reachable.
- **BOTTOM TIM:** This step consists of adding in details of the non-functional requirements of each type of resource using the grid evaluation filled in during the TOP BSM step. Each of these resources is assigned to the corresponding task. A simulation with eBPMN is then performed to display non-functional measurements for the chosen path. If the results are compliant with the non-functional goals to be reached, the implementation phase (TSM) can be carried out. At this step, it is recommended for all analysts to be present, especially when implementation scenario cases are simulated. Several simulation scenarios could be run and evaluated before finding the configuration that satisfies the requirement goals.
- **TSM:** This step describes in more detail how the implementation of a system uses a particular type of resource among those chosen at the BOTTOM TIM step. This step is also required to decide on the precautions to be taken in order to increase the reliability of the process as much as possible. All stakeholders are involved during this stage.
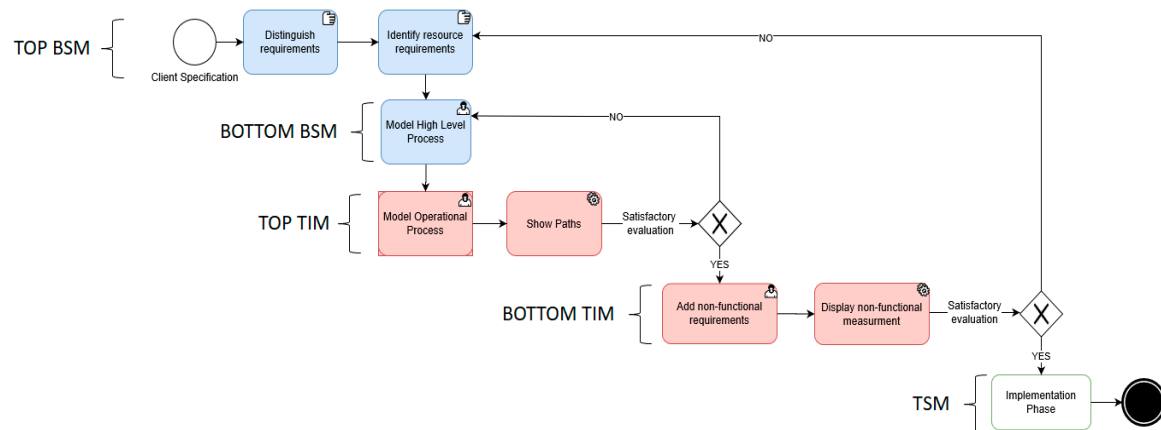
Figure 2. Modeling and simulation methodology based on MDSEA.

3.2. Distinguishing Resource Types

Resources are important for an organization, as they create the product/service provided to the customer (Figure 3). A semantic triangle was proposed by Posada et al. (2004) [26], combining user profile, resources, and model characteristics within a knowledge engineering perspective. This combination helps to better optimize certain industrial processes. From the same perspective, it is recommended by the MDSEA approach to distinguish resources at the first stage of modeling. This work aimed to include these types of resources in the simulation phase at the TSM level in order to assist different stakeholders (business analyst, IT consultants, and users), to facilitate the best decision, and to assign suitable human resources to each task, as well as to choose the adequate physical means and technologies to be used. We describe in the following the characteristics of the three categories of resources:

- **Human resources** or employees are individuals who work to fulfill their tasks according to their roles within a process. Human attributes can be perceived as constraints for the accomplishment of tasks. Their attributes can be defined as overloaded or underloaded, experienced or inexperienced, etc. Employees cannot be repaired or changed, they must adjust their own actions to specific or common constraints as explained in Reference [27]. Thus, human resources' efficiency is not static and varies from time to time depending on the employees' own attributes, their work behavior, etc. Alvaro Segura et al (2020) [28] proposed a conceptual framework showing how the application of different visual computing technologies can empower the human factor.

- **Physical means resources** are tangible items used in a process to carry out its activities. These include a wide range of specific goods and objects depending on the nature of the business, such as machines, robots, or any other physical device. The choice of their efficiency metrics depends on the type of business and on the goal to reach. Generally, the efficiency of physical resources is measured by performance, availability, reliability, etc. [29].

- **Information technology (IT) resources** are everywhere in the business field, and they are considered important enablers of business success and innovation. IT resources are the technological foundation that allows a company to provide real-time, accurate, and comprehensive information for communication. IT is about all the hardware, software, and infrastructure on which IT tools are built. The definition of adequate IT resources depends on the goal to be reached. For example, some focus on hardware performance aspects such as processor usage, while others focus on software performance such as number of licenses, and others focus on infrastructure performance. The metrics used are bit rate, transmission delay, availability, throughput, etc. [30].

Thus, each resource type has its own definition, its own characteristics, and its own indicators of efficiency, hence the importance of distinguishing the type during the modeling and simulation phase [31].
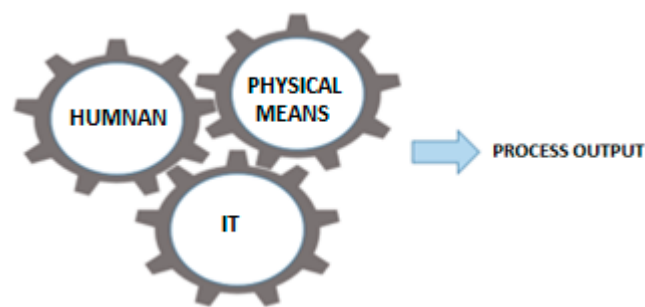
Figure 3. Resource types.

3.2.1. Evaluation Grid

As stated above, the actual work achieved by any process is ultimately done by the resources that perform the tasks making up the process. The goal is to take into consideration the characteristics of each type of resource in order to calculate their efficacy within a process.

To do so, this work proposes the use of an evaluation grid containing the criteria to be taken into account for each resource. The evaluation grid allows an assessment to be made on the quality and performance of a resource that cannot be simply judged good or bad. It is composed of different columns:

- Criterion: The criterion used to evaluate a resource. Each criterion can be divided into sub-criteria;
- Score: A score between 0 and 2 is given to each criterion, with 2 being the best score;
- Weight: The weight of a criterion is intended to capture its relative importance.

The first step consists of choosing the right efficacy measures of each resource. Several characteristics defining the efficacy of human, physical means, and IT resources were provided in the literature. For this work, they were sorted by taking into consideration two criteria:

- Common measurement: we work with several business sectors, so we needed to choose a measurement that is common to all business activities, with the possibility of new ones being added at the client's demand;
- Objective measurement: choosing a clearly defined and easily calculable criterion. Thereby, the stakeholders involved with filling in the grid will reach the same results because they are quantifiable in nature.

Regarding the rating scale, we chose a score between 0 and 2, with 2 being the best score.

The weight of each criterion must reflect its importance regarding the goal to be reached; hence we believe that it makes more sense to let the client choose the weight of each criterion, in order to match their goals.

3.2.2. Extension of the PyBPMN Metamodel

As mentioned in Section 2.3, this work aimed to improve the eBPMN language, as it already implements simulation components for performance analysis. This section briefly described a part of the PyBPMN metamodel used in the eBPMN language. More details are mentioned in References [20,22].

PyBPMN addresses three components to represent a resource in its model:

- The PyBaseResource class represents the resources and is specialized in PyPerformer, PySubSystem, and PyBroker;
- PyPerformer represents unique and atomic resources,
- PySubSystem represents composite resources, groups of resources required to carry out a task,
- PyBroker represents groups of alternative resources that could carry out a task,

In this work, new classes were added to the PyBPMN metamodel to distinguish resource types, as defined in MDSEA. Three specialization classes were added to PyPerformer: PyHuman, PyPhysicalMeans, and PyIT (Figure 4).
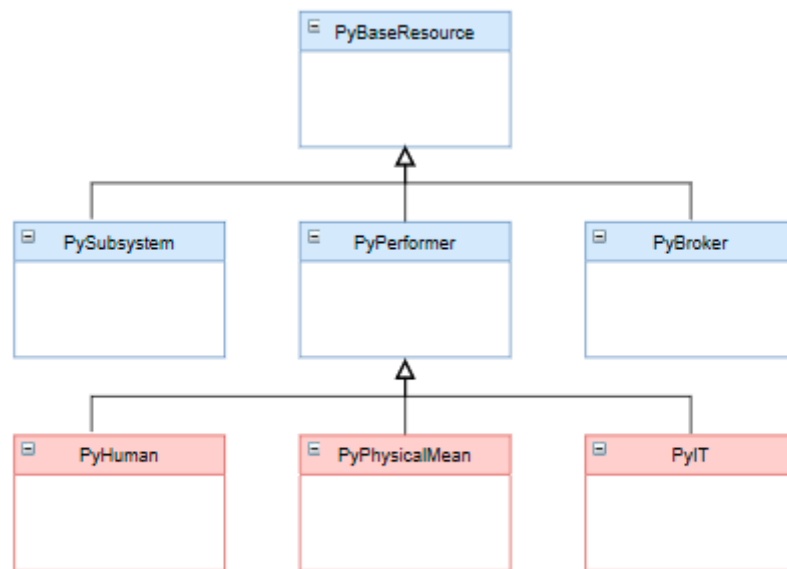
Figure 4. Extension of the PyBPMN metamodel to add the three types of resources.

3.2.3. eBPMN Extension Usage

As reported in Section 2.3, eBPMN uses text annotation of BPMN elements to add non-functional properties according to a formal syntax compliant to Extended Backus-Naur Form (EBNF) as described in Reference [32]. In the example below, a description of the syntax used is detailed and subsequently illustrated in Figure 5:

- The description of an atomic human resource with the name ResponsibleOfMethods with a service time of 100 seconds; the service time is mentioned in PySubSystem, as described in the next point (the same syntax is used for the other types of resources): <<PyHuman>> {name = ResponsibleOfMethods, serviceTime = (value = 1, unit = s)}
- To define a composite task resource with the name of TaskResources containing the three type of resources, ResponsibleOfMethods, ThechnologyA, and MachineA: <<PySubSystem>> {name = TaskResources, components = (ResponsibleOfMethods[100], TechnologyA[10], MachineA[100])}
- The specification that a BPMN task uses the PySubsystem named "TaskResources" is realized through a text annotation associated to the task in the model with the following definition: <<PyDescriptor>> {resources = (TaskResources)}
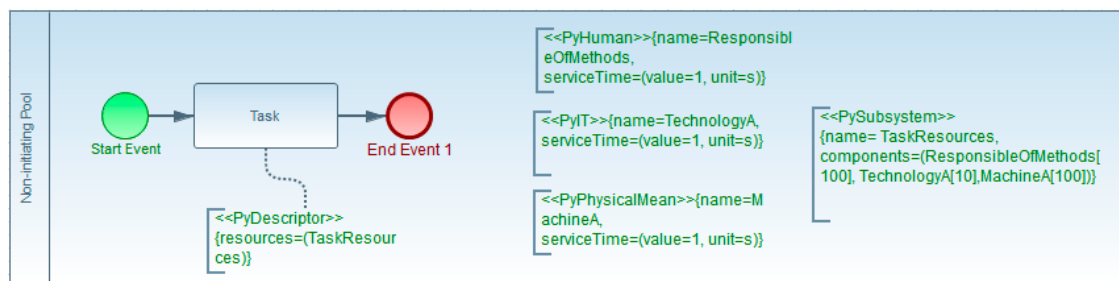

Figure 5. Example of text annotation used for each task.

3.3. Aggregation of Resource Performance

A task could be carried out by multiple resources, and so its performance is related to its resources' performance. Therefore, performance aggregation of resources combined with task description defines the whole task performance.

The resource aggregation consists of describing how resources work together as one coherent, structured entity. Thus, in this case, the performance of the resource aggregation consisted of combining all of the measurements associated to the assigned resources of each task. Resource aggregation can be structured in several forms; in this work, three forms were chosen:

- Sequential (SEQ): All tasks are executed consecutively,
- OO: Only one of the tasks is executed,
- AND: Both tasks are executed simultaneously.

For this study, the task was decomposed into a structured set of atomic subtasks, which meant that each subtask of this decomposition was executed by a single resource (R) as shown in (Figure 6). Performance aggregation then proceeded according to the defined structure shown in Table 1.
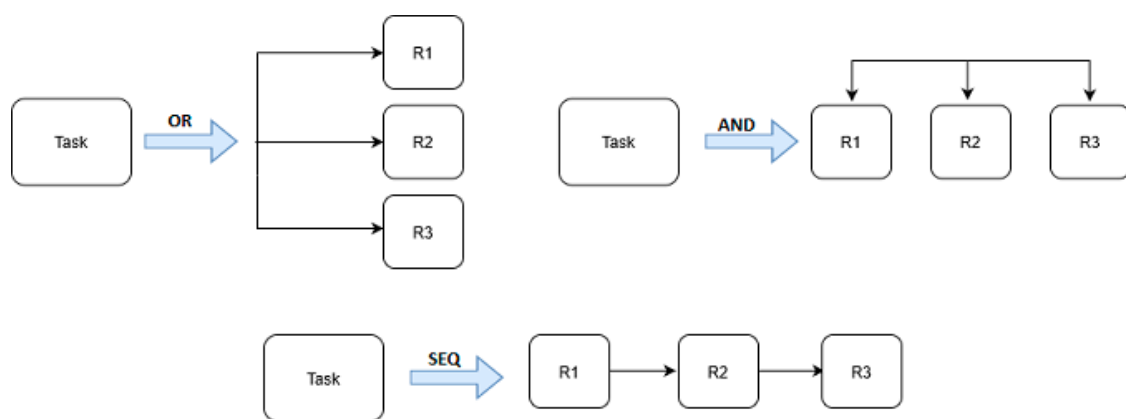

Figure 6. Task decomposition to atomic tasks.

The goal of resource performance aggregation is to obtain a synthetic and global view of the performance and efficiency of the process being proposed. In general, this helps the analyst to choose the most critical process to analyze and to improve. More specifically, within a process, it will help to pinpoint the critical path to enhance. Furthermore, taking into consideration the configuration of the resources that carry out a task allows for more relevant and precise efficacy measurements. As shown in Table 1, each configuration has its own performance aggregation.

To define the structure of the composite resource, the attribute composition=seq/and/or was used and added to the PySubSystem syntax:

<<PySubSystem>> {name = TaskResources, components = (ResponsibleOfMethods[100], TechnologyA[10], MachineA[100]), composition = seq}

3.4. User Interface

Different task types are identified in BPMN to distinguish the specificities of each type. The commonly used tasks are Abstract Task, Service Task, Human, and User Task as described in Reference [33]. User Task represents a task performed by a user with software assistance applications (business process management, enterprise content management …). It is commonly used in BPMN elements and represents a typical workflow task. User tasks are generally performed via an application's user interface, commonly known as the "task form", and are widely used in administrative workflows.

The user interface represents a user-friendly mechanism for interacting with an application. Several researchers have focused on the importance of creating designs to produce the best interface (how to study users and their tasks, human psychology, tools to test a user interface prototype …), and these findings are generally adopted after the simulation phase. To the best of our knowledge, there is no research that provides methodologies or techniques to measure the non-functional performance efficacy of a user interface within the modeling and simulation phase of data processing. This study was intended to contribute to this work, and proposes a composite resource representing the user interface.

Both IT resources and human resources are involved in carrying out a user task, and the efficiency of the operation depends on the efficiency of both resources independently. Since the aim of this work was to provide a finer-granularity simulation, information about the different components used in a user interface was needed (e.g., fields such as single line text, checkbox ...). The idea was to assign to each component the performance attributes linked to the worker who is completing the task form. This proposition aimed to describe the service time that the user requires to fill in each field of a task form (Figure 7) (e.g., to fill a "Department" field for a "Purchase Request" task form, a user might need 5 min)

We believe that this will help stakeholders (analysts, IT consultant and interface designer) to

- Choose the best development techniques to implement the interface user, e.g., to fill some fields automatically or provide some aid that will help the worker in filling them;
- Assign the best worker for a considered task, by taking into consideration workers' experience, their strengths, and their shortcomings when carrying out similar tasks.

Figure 7. User interface example.

3.4.1. PyBPMN Metamodel Extension to Include User Interface Component

To take into account the information about the different components used in a user interface, this work proposes an extension of the PyBPMN metamodel by inheriting the PyInterface class from the PyIT class. The composition link between the PyInterface class and the PyField class is then used, as a task form is composed of one or multiple fields (Figure 8).
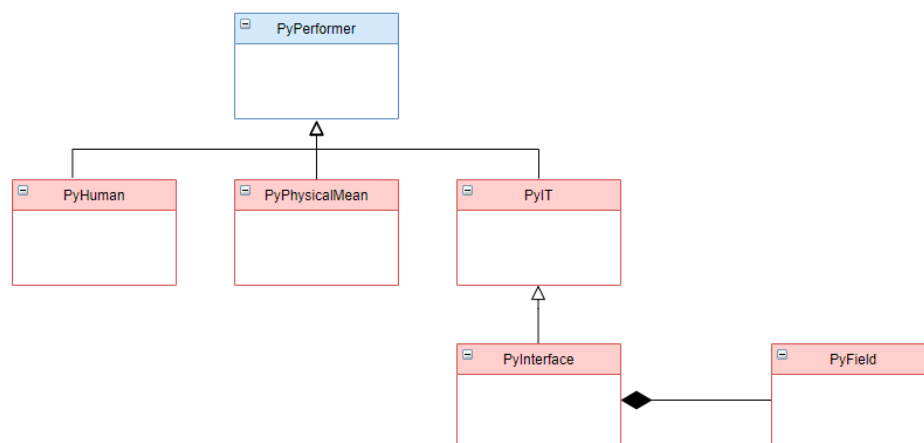

Figure 8. Extension of the PyBPMN metamodel to add composite resources (user and task form).

3.4.2. PyInterface Usage

Below is an example using the proposed syntax. It details the structure and keywords introduced:

- The description of a field of a task with the name "Description" with a service time of 5 min; the service time is mentioned in PyInterface as described in the next point: <<PyField>> {name = Description, serviceTime = (value = 1, unit = min)}
- To define a user interface composite resource with the name of UserInterface: <<PyInterface>> {name = UserInterface, fields = (Name[3], Department[2], RequestData[2], Description[5])}
- The specification that a BPMN user task, uses the PyInterface named "UserInterface" is realized through a text annotation associated to the task in the model: <<PyDescriptor>> {resources = (User_Interface)}

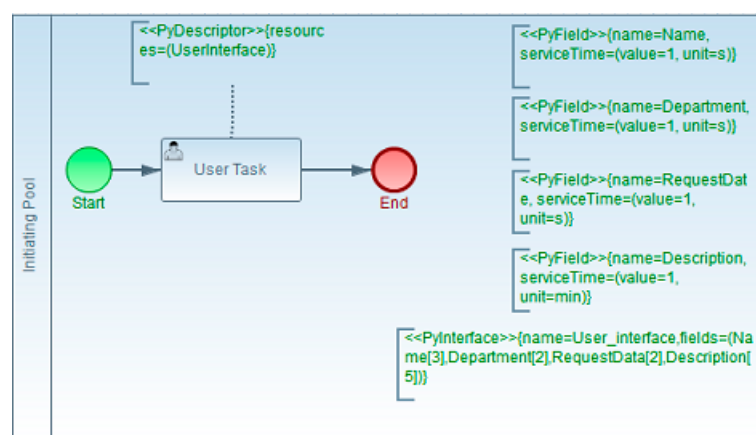Figure 9 illustrates the syntax introduced within the visual modeling tool with a graphical example.

Figure 9. Example of PyInterface in use.

3.5. Performance Measurement of the Process Path

Business process is a sequential flow of activities that can use control gateways to split and merge paths to orientate the flow. It splits the activity into different possible flows and hence produces multiple alternatives for execution paths [15]. Thus, process paths represent all the business process cases users are likely to execute during runtime. Depicting the process path's details helps the stakeholders to better analyze it by more easily identifying errors (missing task, sequence flow error, etc.). Furthermore, the analysis of the impact of the different alternatives of execution scenarios of a model is best done by displaying the performance measurements of a process path, rather than a set of isolated tasks. This makes the investigation more global and efficient by identifying weak areas, which facilitates the possibility of focusing on the potential solutions for process improvement. For example, having the waiting time of a path gives more insights on how to improve the waiting time of the whole process. The same concept applies to any non-functional properties.

Moreover, at runtime, only a small selection of process paths is executed from the several available. Therefore, process execution time, reliability, and cost are linked to the most executed process cases.

The goal of this work was to give an outline of the performance and reliability of the paths chosen by the business analyst and the stakeholders. The choice could be made by using probabilities, as eBPMN already allows, or by explicitly specifying the user choice. For example, a process could lead to path A if the user approves the task, or lead to path B if the user refuses the task. The business analyst could choose the performance path to display by using the user choice technique "approve/refuse", or by giving a probability to each path of A: 50% and B: 50%.

Thus, the business analysts and stakeholders could concentrate their efforts on finding solutions for the most likely process cases that can be executed (eliminating unnecessary steps, increasing resources, decreasing failure activities, etc.).

To do so, the eBPMN language uses the token concept for BPMN simulation. The token demonstrates the underlying behavior of each BPMN element. It traverses a sequential flow from the start point to the end point of a process [15]. Thus, a token could be considered a reference of a process instance, and it is used to display the different elements it went through.

In the eBPMN implementation, simulation entities are categorized into two types:

- eBPMN objects corresponding to the original BPMN metamodel,
- eBPMN objects derived from the PyBPMN metamodel.

This work used the first type to display the different paths through which the token went and used the second one to display the performance measurement of each path after the simulation.

To do so, a path class was added to the eBPMN class diagram. We have provided a brief description below, with more details described in Reference [34].

- EBPMN ELEMENT class: is the superclass of every simulation entity of the Ebpmn implementation. It is specialized into two further general classes, FlowNode and EBPMNResource classes;
- EBPMNResource class: represents the entities corresponding to the execution behavior of resource elements according to the PyBPMN metamodel as shown in Figure 4;
- FlowNode class: represents the entities corresponding to the execution behavior of BPMN FlowNode elements.

For our purpose, since a path is a set of BPMN elements, we extended the eBPMN class diagram by adding a composition link between the "FlowNode" class and the "Path" class (Figure 10).
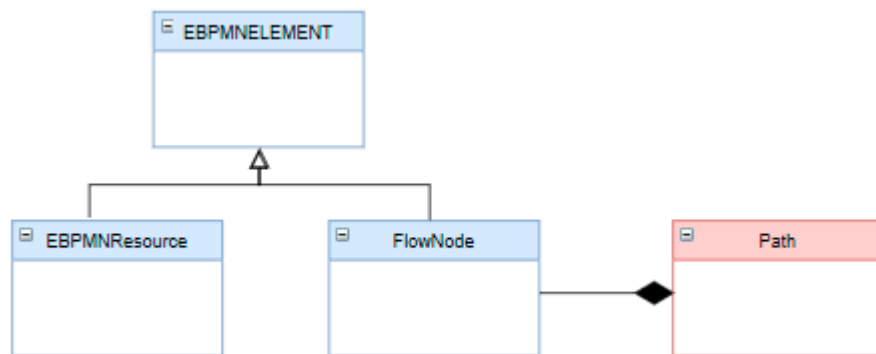
Figure 10. eBPMN extension with PATH class.

Figure 11 illustrates an example of the path choice techniques. The business analyst could choose the path using path probabilities (A in Figure 11, already possible using eBPMN), or could choose the path using operational requirements (B in Figure 11) by using the attribute userchoice = "choice" in PyDescription syntax: <<PyDescriptor>> {resources = (evaluator[720]), userChoice = Validated}.

They could also combine the two methods within a process.
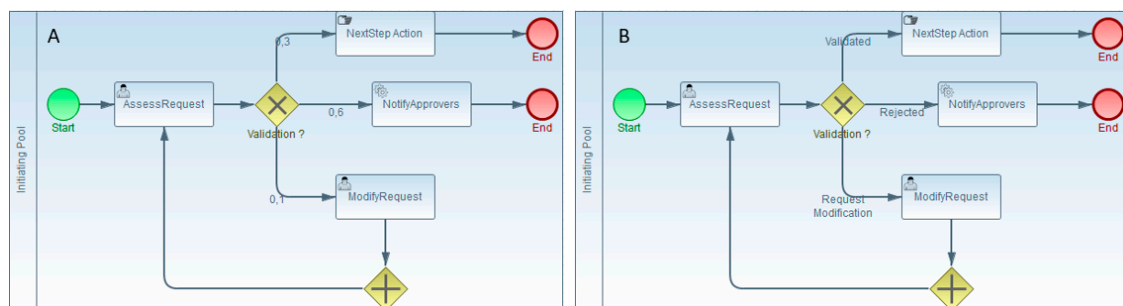
Figure 11. Path choice techniques.

3.6. Case Study

This section aims to demonstrate the simulation feasibility of the proposed methodology and eBPMN extension. For the sake of simplicity, the study presented reflects some aspects of the simulation performed on a real case. A workflow extracted from one of Exakis' client models was used for this example. Additionally, only two profiles of human resources were considered (experienced and inexperienced resources). In order to preserve our client's confidentiality, we have called it "Brand". Brand is a French fast food chain that mainly sells sandwiches and salads, and that is expanding across France. Due to this expansion, they need to improve their workflow to meet market expectations.

3.6.1. Top BSM

The first step in this stage consists of distinguishing conceptually functional and non-functional requirements.

➢ **Functional requirements**: In order to open new franchise stores, maintain, or close the existing ones, Brand wants to use a workflow to automate the evaluation of the methods and practices used in those stores. The evaluation is based on qualitative criteria assessed by a qualified evaluator. The request is then submitted for validation to a validation circuit.

➢ **Non-functional requirements**: Their principal goals are
- To increase internal efficiency for managing opening and closing requests,
- To be able to process more requests while improving response and decision delay,
- Late closing of an inadequate franchised store could be harmful to their image,
- Late opening of a good potential store could decrease their sales.

They receive on average five requests per week and, in order to reach their goals, their target is to process two requests a week.

Following this is the identification of resource's requirements using grid evaluation, as explained in Section 3.2.1. Table 2 shows an example with two evaluators' profiles.

Table 2. Grid evaluation of human resource characteristics.

Profile of an Evaluator	Availability		Skills				Performance				Results
	Workload		Business Skill		Computer Skill		Speed of Execution		Task Accuracy		
	Score	Weight	Score	Weight	Score	Weight	Score	Weight	Score	Weight	
Experienced Evaluator	1	2	2	2	2	1	2	4	2	2	1.82
Inexperienced Evaluator	1	2	0	2	1	1	0	4	2	2	0.64

3.6.2. Bottom BSM

This stage consists of formally modeling the process at a high level of abstraction using the BPMN language. Resources are distinguished using the different task types provided by BPMN, as shown in Figure 12.
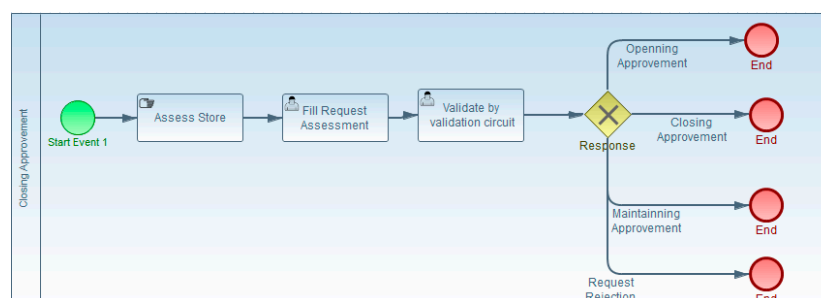
Figure 12. High-level process modeling.

3.6.3. Top TIM

The first step of this stage is to specify the functional requirements in detail, using the BPMN language to describe activities and their flow sequence (Figure 13).

The second step of this stage is to carry out a path verification to check whether the desired paths, as explained in Section 3.5, are reachable and compliant with the functional requirements. As shown in Figure 14, Path 1 was reachable—the token was able to reach the end statement—while in Path 2 it could not. Before moving on to the next phase, it is important to make sure that all the desired study paths are accessible.

Figure 13. Operational process modeling.

```
================================================================

                        PATH REACHABILITY

================================================================
================================================================

    1. Start==>AssessStore==>FillTemplateFile==>FillEvaluationRequest==>SaveDataEv
       aluation==>ValidateRequestMethods==>SaveDataMethods==>RequestQualityValidat
       ion==>SaveDataQuality==>CheckLegalConditions==>SaveDataALegal==>AssessLegal
       Conditions==>SaveDataRLegal==>ValidateRequestEvaluation==>SaveDataMFranchis
       e==>NotifyEvaluatorAndOperators==>End

    2. Start==>AssessStore==>FillTemplateFile==>FillEvaluationRequest==>SaveDataEv
       aluation==>ValidateRequestMethods==>SaveDataMethods==>RequestQualityValidat
       ion==>

================================================================
```
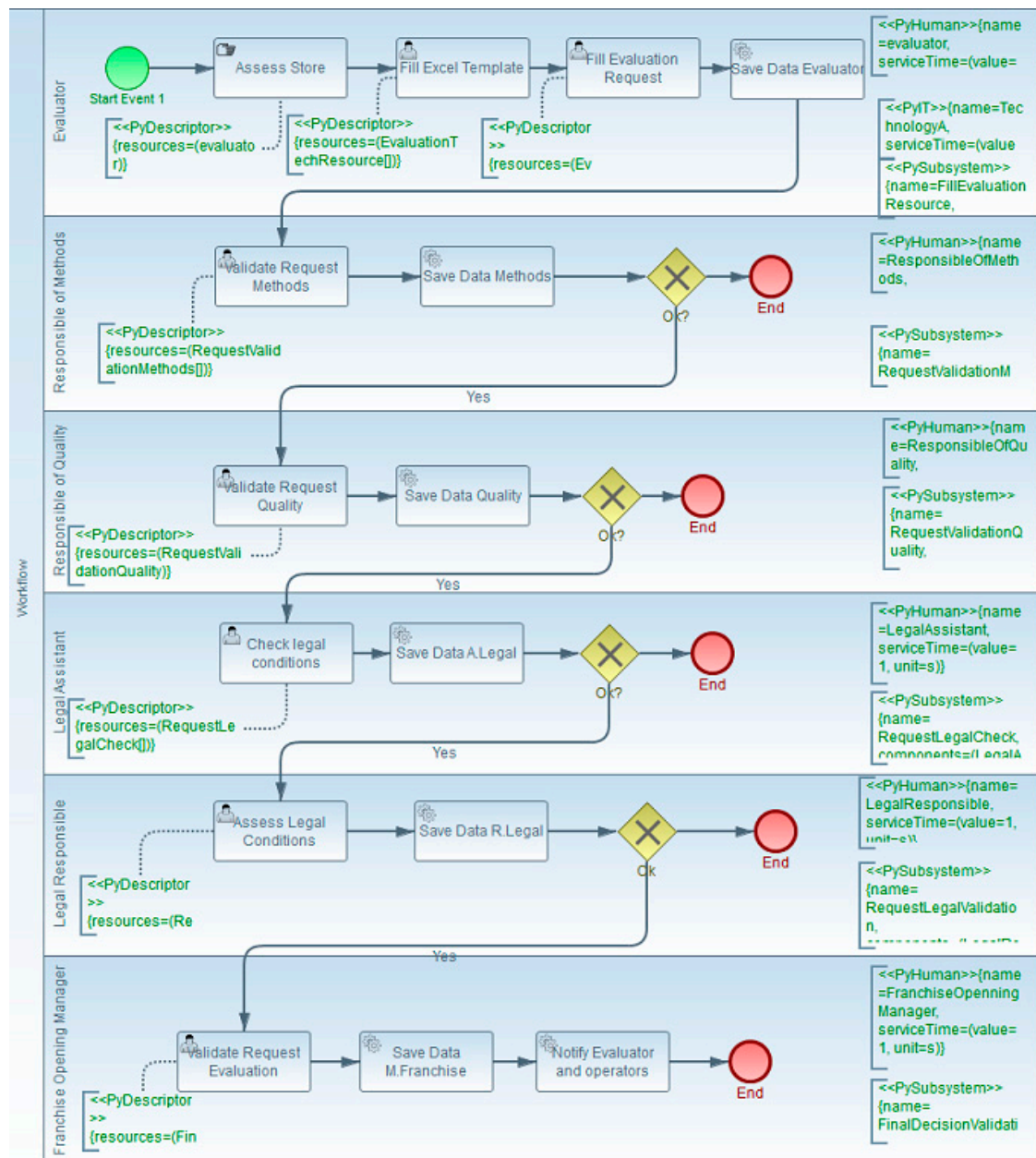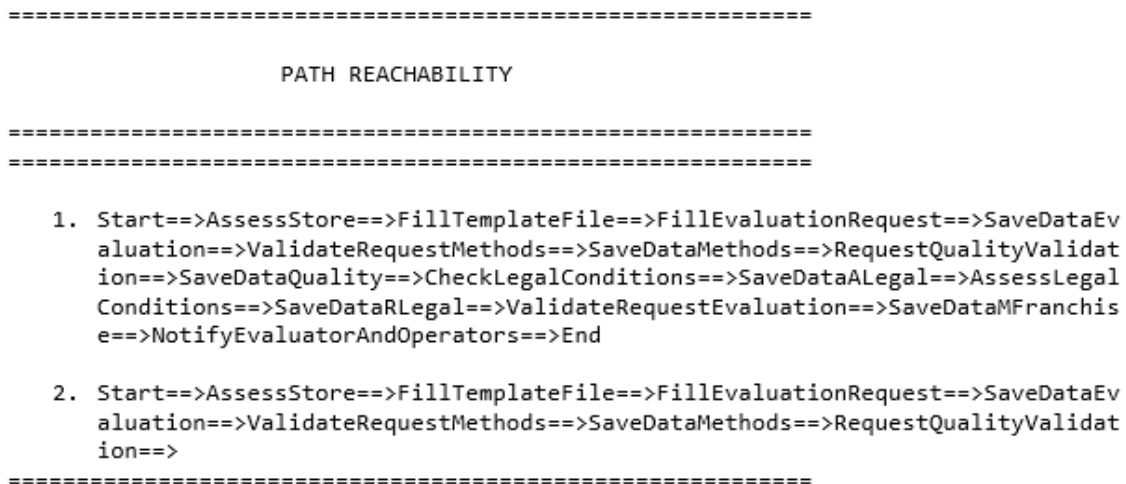
Figure 14. Path reachability.

3.6.4. Bottom TIM

The first step of this phase is to determine all the non-functional measurements of each task using the grid evaluation, as explained in Section 3.2.1. To maintain the simplification, only the evaluator's tasks are shown in Table 3.

Table 3. Definition of non-functional measurements of each task.

Tasks	Average Execution Time	Experienced Eval	Inexperienced Eval
Assess Store	720	396	1125
Fill Excel Template	61	34	95
Fill Evaluation Request	31	17	48

The second step of this phase is to associate each task with its resources, their structure, and their non-functional measurements (Figure 15), as explained in Sections 3.2.3 and 3.3. In this case study, the structure of all composite resources was parallel.
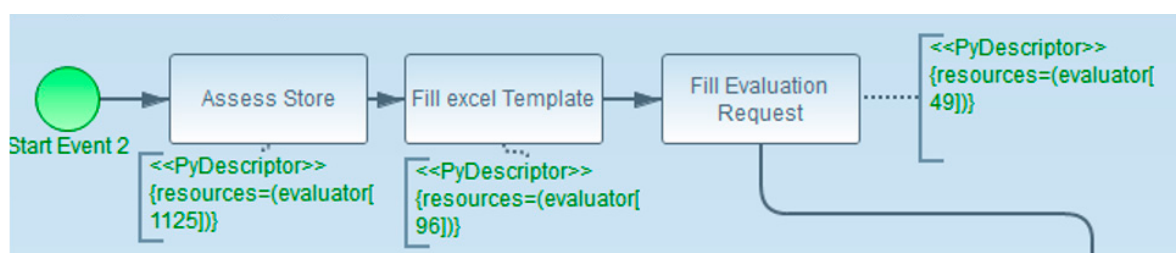

Figure 15. Tasks associated with non-functional measurements.

Three ways to display the results are proposed:

- Getting global results using resource aggregation: processing time of each path the simulation ran using the resource aggregation technique,
- Getting global results including the type of each resource: processing time of each path the simulation ran through, and the processing time of each type of results,
- Getting detailed results: processing time of each task and each resource on their own.

Only the results of the first two options are shown, since this was the proposed eBPMN enhancement. Two case scenarios were considered, one with experienced human resources and the

other with inexperienced ones. The duration of simulation used was 30 days, with five requests launched per day.

Global Results Using Resource Aggregation

For the first step of the non-functional simulation phase, it was decided to begin with the worst case scenario (inexperienced human resource) to investigate unnecessary steps that could be removed.

As shown in Figure 16, in this scenario, a request was processed within 10 days, which was far from the goal. After process analysis, it was found that the "Check legal conditions" and "Assess legal conditions" steps were quite similar, so it was decided to remove the "Check legal conditions" activity. The simulation results of the new configuration showed that a request was processed within 8 days (Figure 17). Several scenarios were trialed until only the important tasks remained. It was important to check the path reachability after each configuration.

```
============================================================

                PATH STATISTICS: RESOURCE AGGREGATION

============================================================
### Path.................... :

Start Event 1===>Assess Store===>Fill Excel Template==>Fill Evaluation
Request==>Save Data Evaluation==>Validate Request Methods==>Save Data
Methods==>Validate Request Quality==>Save Data Quality==>Check legal
conditions==>Save DataA Legal==>Assess Legal Conditions==>Save DataR
Legal==>Validate Request Evaluation==>Save DataM Franchise==>Notify Evaluator and
operators==>End
Mean Service Time ...................: 8780.753 min
Mean Waiting Time ...................: 5639.485 min
============================================================
```

Figure 16. Global results using resource aggregation.

```
==============================================================

                PATH STATISTICS: RESOURCE AGGREGATION

==============================================================
### Path.................... :

Start Event 1==>Assess Store==>Fill Excel Template==>Fill Evaluation
Request==>Save Data Evaluation==>Validate Request Methods==>Save Data
Methods==>Validate Request Quality==>Save Data Quality==>Assess Legal
Conditions==>Save DataR Legal==>Validate Request Evaluation==>Save DataM
Franchise==>Notify Evaluator and operators===>End
Mean Service Time ...................: 4130.492 min
Mean Waiting Time ...................: 7471.699 min
==============================================================
```

Figure 17. Global results for inexperienced human resource after removing the "Check Legal conditions" step.

After deleting unnecessary steps, the best case scenario simulation was launched (simulation with experienced human resources). As shown in Figure 18, the results showed that a request was processed within 6 days. To gain better insights on which type of resource needed improvements, a simulation showing the results for each type was carried out, as shown in step "Global Results Showing the Results of Each Type of Resource" below.

```
============================================================
                PATH STATISTICS: RESOURCE AGGREGATION

============================================================
### Path................... :

Start Event 1==>Assess Store==>Fill Excel Template==>Fill Evaluation
Request==>Save Data Evaluation==>Validate Request Methods==>Save Data
Methods==>Validate Request Quality==>Save Data Quality==>Assess Legal
Conditions==>Save DataR Legal==>Validate Request Evaluation==>Save DataM
Franchise==>Notify Evaluator and operators==>End
Mean Service Time ...................: 2742.330 min
Mean Waiting Time ...................: 5635.648 min
============================================================
```

Figure 18. Global results for experienced human resource using resource aggregation.

It is important to mention that according to References [24,25] and as described in Section 2.4, for processing time, the maximum time between the parallel measurements was selected. This proved the importance of taking into consideration the structure of the resource composition. Supposing that the simulation of resources is done only sequentially, double the result will be obtained, as shown in Figure 16, whilst the opposite is also true. Thus, taking into account the structure of resource composition avoids unnecessary changes being made.

Global Results Showing the Results of Each Type of Resource

In this step, a simulation was carried out from the modified model, as seen in step "Global Results Using Resource Aggregation" above. Two scenarios (experienced and inexperienced human resources) were simulated to analyze which type of resource needed improvement.

As the simulation results in Figures 19 and 20 showed for the two scenarios, IT resources had a longer service time. Since the best human-type scenario was already launched, improving the IT-type may be significant especially considering that human resources and IT resources work simultaneously. Thus, it is important to enhance IT resources before making any changes in human ones.

```
============================================================
                PATH STATISTICS: RESOURCE TYPE DETAILS

============================================================

============================================================
### Path................... :
Start Event 1==>Assess Store==>Fill Excel Template==>Fill Evaluation
Request==>Save Data Evaluation==>Validate Request Methods==>Save Data
Methods==>Validate Request Quality==>Save Data Quality==>Assess Legal
Conditions==>Save DataR Legal==>Validate Request Evaluation==>Save DataM
Franchise==>Notify Evaluator and operators==>End
Mean Service Time ...................: 8800.168 min
Mean Waiting Time ...................: 5700.890 min
============================================================
### Resources Path................... :

Type Resources................... :

Human Resources:
*************************************
Mean Service Time ...................: 7900.133 min
Mean Waiting Time ...................: 4000.226 min
IT Resources:
*************************************
Mean Service Time ...................: 8000.35 min
Mean Waiting Time ...................: 5700.890 min


============================================================
```

Figure 19. Global results of each type of resource for inexperienced human resource.

```
============================================================

                PATH STATISTICS: RESOURCE TYPE DETAILS

============================================================


============================================================
### Path.................... :
Start Event 1==>Assess Store==>Fill Excel Template==>Fill Evaluation
Request==>Save Data Evaluation==>Validate Request Methods==>Save Data
Methods==>Validate Request Quality==>Save Data Quality==>Assess Legal
Conditions==>Save DataR Legal==>Validate Request Evaluation==>Save DataM
Franchise==>Notify Evaluator and operators==>End
Mean Service Time ....................: 2680.168 min
Mean Waiting Time ....................: 5300.890 min
============================================================
### Resources Path.................... :


Type Resources.................... :


Human Resources:
****************************************
Mean Service Time ....................: 2000.48 min
Mean Waiting Time ....................: 5200.226 min
IT Resources:
****************************************
Mean Service Time ....................: 2680.168 min
Mean Waiting Time ....................: 5300.890 min


============================================================
```

Figure 20. Global results of each type of resource for experienced human resources.

Distinguishing the results of each type of resource helps to gain more information into how resources are involved within a process, which eases the process of decision-making.

User Interface

A finer simulation was carried out using the PyInterface composite resource to assess the efficacy of user interfaces, which improves IT resources' service times. After the first attempt of running the simulation using the example user interface shown in Figure 21, the global performance of the legal opinion task was equal to 3544 min. The analysis of the simulation results highlighted that some fields could be filled automatically and on-hand helpful information could be provided to facilitate the validation of the task, which was optimized to 1714 min, as shown in the second attempt depicted in Figure 22. Table 4 shows some interface fields and their improvements.

Table 4. Interface fields descriptions and improvements.

Field	Filling Time	Description	Improvement	Filling Time
Responsible of Legal opinion (Text Field)	15 s	The name is filled manually, filling it automatically saves time	Filled automatically	0 s
Evaluation Type (Text field)	15 s	The evaluation type is filled manually, picking a choice instead would save time	Choice field	6 s
Sales point type (Text field)	20 s	The sales point type is filled manually, picking a choice instead would save time	Choice field	6 s
Region (Choice field)	7 s			
City (Choice field)	20 s	Filter the city field depending on the chosen region would save time	Choice field filtered by region	10 s

Table 4. *Cont.*

Field	Filling Time	Description	Improvement	Filling Time
Evaluation Analysis	3520 min	For the evaluation, the supervisor needs to compare this request with previous requests using the same criteria, and also check legal points with the same criteria. To do so they need to search for them manually in the system, which could be time consuming, especially if it is a complex request. For improvement on hand information is displayed on demand depending on the chosen criteria, thus allowing the supervisor to focus on the analysis	On hand Information	1800 min

Figure 21. User interface results.

Figure 22. User interface results after improvement.

After investigation into each user interface used within the process, a global simulation was carried out to assess the new configurations. The simulation results shown in Figure 23 showed that a request was processed within 3 days, which was closer to the desired goal. Alternative configurations were evaluated by repeating the same steps until a configuration that satisfies the requirement goals was achieved.

Figure 23. Global results after user interface improvements.

4. Discussion

The definition of resources in the BPMN 2.0 standard is abstract. It is clearly stated in its specification that the modeling of organizational structures and resources is outside of the scope, as mentioned in Reference [15]. Despite the relevance of the studies proposed to model resource specification and resource management, they show only some aspects of how a resource is used within a workflow case.

There are two types of contributions in resource simulation in BPMN. The first type of contribution enhances the expressive capabilities of BPMN in terms of human resource technical behavior (allocation, distribution ...), in order to select the best technology/WFMS to support the organizational needs, as in References [21,35,36]. The second type of contribution highlights the importance of considering the non-functional requirements of resources rather than only the control and data flow, since the non-functional requirements of a business model are affected by the skills and other factors of the resources assigned, as in References [22,37,38]. This work belongs to the second category and builds on the research of Bocciarelli et al. [22]. To the best of our knowledge, their proposition is the only one that introduced the composite resource whilst respecting the BPMN extension recommendations.

This work distinguished atomic and composite resource types. As described previously, each type has its own characteristics to be taken into account when computing its service time and its reliability. For example, to calculate the service time of a human resource, we should take into account their skills, experience, availability, etc. For IT resources, however, throughput and loading time are more important to take into consideration. In addition, these parameters have positive or negative impacts on the non-functional indicator. An unavailable and less skilled worker will take more time to execute a task and vice versa. This shows that taking these characteristics into account will help the right questions to be asked regarding the decisions to be made to improve a process.

The goal of resource performance aggregation is to obtain a synthetic and global view of the performance efficacy situation. In general, it helps the analyst to choose the most critical process for analysis and enhancement. More specifically, within a process, it will help to quickly point out the critical path that can be improved. Furthermore, the computation of parallel, sequential, and alternative structures are very different. Thus, taking into consideration the structure of the resources that carry out a task allows for more precise and reliable measures of efficacy, and could avoid unnecessary changes being made.

After carrying out an overall simulation, this work proposes the use of composite resources (worker and task form). The idea is to assign to each field of a task form the service time the user takes to fill it. This finer-granularity simulation helps to better design graphical interfaces by providing only the information needed to save time and to decrease the error rate. That is very helpful for administrative processes for which user interfaces are their main core.

The non-functional indicators used in this work were limited to performance and reliability. For future work, adding a more exhaustive indicator would be interesting and might better meet the requirements. Firstly, we believe that adding cost as a non-functional indicator would be helpful and add value to decision-making in order to reach process effectiveness. Cost represents the spending required for the execution of a business process. Thus, performance, reliability, and cost are fully interrelated. The amount of time/quantity required to perform/repair tasks is directly related to the cost of resources assigned. Therefore, the analyst could have a global and complete idea of the costs of the different simulation scenarios played. This is even more important when we take into consideration the characteristics of each type of resource; an experienced worker is more expensive than an inexperienced one.

Furthermore, the relationship between performance and reliability was been considered in this work. Finding the association between them will help to enlighten cause–effect connections, which provide additional decision-making information. To do so, some research used Bayesian networks [39] or AHP [40], but encountered some difficulties in getting information from decision-makers. From our literature review, the best way to address this limitation would be to use mathematical

techniques using a lot of data (big data). Additionally, a recent study on eBPMN as described in Reference [41] used a brand-new approach where the reliability indicator was directly simulated, which increased its relationship with the time service indicator. This work could be the first path to explore, as our methodology was based on this language. Finally, another interesting possible future line of research would be the visualization of aforementioned concepts and their relationships in the domain of the process flow from a knowledge sharing or information management perspective as explained in Reference [42]. For instance, using text annotation to represent additional needs is not user friendly, so it might not gain wide acceptance and it would also decrease the readability of complex BPMN models. Hence, it is important to find a mechanism that allows the use of graphic components to add non-functional requirements while remaining compliant with BPMN specification instructions.

5. Conclusions

This paper presents a data process automation methodology based on the MDSEA approach, using BPMN and eBPMN languages for M&S. Visual modeling allows a more detailed and accurate setting of resource involvement by separating and taking into consideration the resource structure of human, IT, and physical means. Visual modeling interfaces and annotation components developed in this work allow the definition of new composite resources including user (human resource) and task forms (IT resource) in order to obtain finer-granularity simulations. The simulation proposed in this work will help analysts and stakeholders to analyze and predict the behavior of the process following several alternative actions before making decisions on their implementation. It will help to quantify the time needed to call and use different resources involved within a process case. As a result, it could help the best decision to be made, as well as helping to simplify and/or remove infrequently used paths.

It is assumed that the more progress is made during solution design, the more detailed and specific information must be modeled and simulated before its implementation. With this in mind, this approach contributes to providing detailed specifications to anticipate and reduce the risk of creating a new process or improving a process implementation at the execution phase. However, the approach is still very dependent on available information at this early design stage.

References

1. Alcácer, V.; Cruz-Machado, V. Scanning the Industry 4.0: A Literature Review on Technologies for Manufacturing Systems. *Eng. Sci. Technol. Int. J.* **2019**, *22*, 899–919. [CrossRef]
2. Weske, M. *Business Process Management: Concepts, Languages, Architectures*; Springer: Berlin/Heidelberg, Germany, 2019; ISBN 978-3-662-59431-5.
3. Rodrigues da Silva, A. Model-driven engineering: A survey supported by the unified conceptual model. *Comput. Lang. Syst. Struct.* **2015**, *43*, 139–155. [CrossRef]
4. Earnshaw, R. Visual Computing. In *Data Science and Visual Computing*; Advanced Information and Knowledge Processing; Springer International Publishing: Berlin/Heidelberg, Germany, 2019; pp. 33–38, ISBN 978-3-030-24366-1.

5. Posada, J.; Toro, C.; Barandiaran, I.; Oyarzun, D.; Stricker, D.; de Amicis, R.; Pinto, E.B.; Eisert, P.; Dollner, J.; Vallarino, I. Visual Computing as a Key Enabling Technology for Industrie 4.0 and Industrial Internet. *IEEE Comput. Grap. Appl.* **2015**, *35*, 26–40. [CrossRef] [PubMed]

6. Zacharewicz, G.; Pirayesh-Neghab, A.; Seregni, M.; Ducq, Y.; Doumeingts, G. Simulation-Based Enterprise Management. In *Guide to Simulation-Based Disciplines*; Mittal, S., Durak, U., Ören, T., Eds.; Simulation Foundations Methods and Applications; Springer International Publishing: Cham, Switzerland, 2017; pp. 261–289, ISBN 978-3-319-61263-8.

7. Bocciarelli, P.; D'Ambrogio, A.; Giglio, A.; Paglia, E.; Gianni, D. Empowering Business Process Simulation through Automated Model Transformations; Spring Simulation Multiconference. In Proceedings of the Symposium on Theory of Modeling & Simulation—DEVS Integrative, Tampa, FL, USA, 13–16 April 2014.

8. D'Ambrogio, A.; Zacharewicz, G. Resource-based Modeling and Simulation of Business Processes. In Proceedings of the 2016 Summer Computer Simulation Conference (SCSC 2016), Society for Modeling and Simulation International (SCS), Montreal, QC, Canada, 24–27 July 2016.

9. Dumas, M.; La Rosa, M.; Mendling, J.; Reijers, H.A. *Fundamentals of Business Process Management*; Springer: Berlin/Heidelberg, Germany, 2018; ISBN 978-3-662-56508-7.

10. Ducq, Y. An Architecture for Service Modelling in Servitization Context: MDSEA. In *Enterprise Interoperability*; Zelm, M., van Sinderen, M., Pires, L.F., Doumeingts, G., Eds.; John Wiley & Sons, Ltd: Chichester, UK, 2014; pp. 1–19, ISBN 978-1-118-84699-5.

11. Schmidt, D.C. Guest Editor's Introduction: Model-Driven Engineering. *Computer* **2006**, *39*, 25–31. [CrossRef]

12. Drozdova, M.; Kardos, M.; Kurillova, Z.; Bucko, B. Transformation in Model Driven Architecture. In *Information Systems Architecture and Technology: Proceedings of 36th International Conference on Information Systems Architecture and Technology—ISAT 2015—Part I*; Borzemski, L., Grzech, A., Świątek, J., Wilimowska, Z., Eds.; Advances in Intelligent Systems and Computing; Springer International Publishing: Cham, Switzerland, 2016; pp. 193–203, ISBN 978-3-319-28553-5.

13. Bourey, J.-P.; Grangel, R.; Ducq, Y. *Report on Model Driven Interoperability*; INTEROP Interoperability Research for Networked Enterprises Applications and Software: Lille, France, 2007.

14. Rolstadås, A.; Andersen, B. *Enterprise Modeling*; Springer US: Boston, MA, USA, 2000; ISBN 978-1-4613-7016-1.

15. Object Management Group. *Business Process Model and Notation (BPMN)—Version 2.0*. 2011. Available online: https://www.omg.org/spec/BPMN/2.0 (accessed on 29 June 2020).

16. van der Aalst, W.M.P. Business Process Simulation Survival Guide. In *Handbook on Business Process Management 1*; vom Brocke, J., Rosemann, M., Eds.; Springer: Berlin/Heidelberg, Germany, 2015; pp. 337–370, ISBN 978-3-642-45099-0.

17. Korherr, B.; List, B. Extending the EPC with performance measures. In *Proceedings of the 2007 ACM Symposium on Applied Computing—SAC '07*; ACM Press: Seoul, Korea, 2007; p. 1265.

18. Gagne, D.; Trudel, A. Time-BPMN. In *Proceedings of the 2009 IEEE Conference on Commerce and Enterprise Computing*; IEEE: Vienna, Austria, 2009; pp. 361–367.

19. Saeedi, K.; Zhao, L.; Sampaio, P.R.F. Extending BPMN for Supporting Customer-Facing Service Quality Requirements. In *Proceedings of the 2010 IEEE International Conference on Web Services*; IEEE: Miami, FL, USA, 2010; pp. 616–623.

20. Bocciarelli, P.; D'Ambrogio, A. A BPMN Extension for Modeling Non Functional Properties of Business Processes. In Proceedings of the Spring Simulation Multi-conference, SpringSim; ACM, Boston, MA, USA, 3–7 April 2011.

21. Stroppi, L.J.R.; Chiotti, O.; Villarreal, P.D. Defining the resource perspective in the development of processes-aware information systems. *Inf. Softw. Technol.* **2015**, *59*, 86–108. [CrossRef]

22. Bocciarelli, P.; D'Ambrogio, A.; Giglio, A.; Paglia, E. A BPMN Extension to Enable the Explicit Modeling of Task Resources; CEUR Workshop Proceedings. In Proceedings of the INCOSE Italia Conference on Systems Engineering, Turin, Italy, 14–16 November 2016; Volume 1728, pp. 40–47.

23. Heguy, X.; Zacharewicz, G.; Ducq, Y.; Tazi, S.; Vallespir, B. A Performance Measurement Extension for BPMN: One Step Further Quantifying Interoperability in Process Model. In *Enterprise Interoperability VIII*; Popplewell, K., Thoben, K.-D., Knothe, T., Poler, R., Eds.; Proceedings of the I-ESA Conferences; Springer International Publishing: Cham, Switzerland, 2019; pp. 333–345, ISBN 978-3-030-13692-5.

24. Cardoso, J.; Sheth, A.; Miller, J.; Arnold, J.; Kochut, K. Quality of service for workflows and web service processes. *J. Web Semant.* **2004**, *1*, 281–308. [CrossRef]

25. Ducq, Y.; Vallespir, B. Definition and aggregation of a performance measurement system in three aeronautical workshops using the ECOGRAI method. *Prod. Plan. Control* **2005**, *16*, 163–177. [CrossRef]

26. Posada, J.; Wundrak, S.; Stork, A.; Toro, C. Semantically Controlled LMV Techniques for Plant Design Review. In *Proceedings of the Volume 4: 24th Computers and Information in Engineering Conference*; ASMEDC: Salt Lake City, UT, USA, 2004; pp. 329–335.

27. Rangra, S.; Sallak, M.; Schon, W.; Vanderhaegen, F. A Graphical Model Based on Performance Shaping Factors for Assessing Human Reliability. *IEEE Trans. Rel.* **2017**, *66*, 1120–1143. [CrossRef]

28. Segura, Á.; Diez, H.V.; Barandiaran, I.; Arbelaiz, A.; Álvarez, H.; Simões, B.; Posada, J.; García-Alonso, A.; Ugarte, R. Visual computing technologies to support the Operator 4.0. *Comput. Ind. Eng.* **2020**, *139*, 105550. [CrossRef]

29. Moore, P.V.; Briken, K.; Engster, F. Machines & Measure. *SSRN J.* **2020**, *44*, 139–144.

30. Dodaro, G.L. *GAO/AIMD-98-89 IT Performance Measurement*; United States General Accounting Office, 1998. Available online: https://www.gao.gov/special.pubs/ai98089.pdf (accessed on 29 June 2020).

31. Ougaabal, K.; Zacharewicz, G.; Ducq, Y.; Tazi, S. Distinguishing Resource Type in Bpmn Workflows at Simulation Phase. In *Proceedings of the Theory and Foundations of Modeling and Simulation (TMS 2019)*; Society for Modeling and Simulation International (SCS): Tucson, AZ, USA, 2019.

32. Scowen, R.S. *Extended BNF—A Generic Base Standard*; Software Engineering Standards Symposium, 1998; Volume 3, pp. 2–6. Available online: https://www.cl.cam.ac.uk/~{}mgk25/iso-14977.pdf (accessed on 29 June 2020).

33. Polancic, G. BPMN 2.0 Task Types Explained. Available online: https://fr.slideshare.net/JoevanSantos/bpmn-20tasktypesexplained (accessed on 29 June 2020).

34. Paglia, E. Innovative Approaches for Business Process Modeling and Simulation. Ph.D. Thesis, University of Rome "Tor Vergata", Rome, Italy, 2017.

35. Russell, N.; van der Aalst, W.M.P.; ter Hofstede, A.H.M.; Edmond, D. Workflow Resource Patterns: Identification, Representation and Tool Support. In *Active Flow and Combustion Control 2018*; King, R., Ed.; Notes on Numerical Fluid Mechanics and Multidisciplinary Design; Springer International Publishing: Berlin/Heidelberg, Germany, 2005; Volume 141, pp. 216–232, ISBN 978-3-319-98176-5.

36. Awad, A.; Grosskopf, A.; Meyer, A.; Weske, M. *Enabling Resource Assignment Constraints in BPMN*; Technical Report; Hasso Plattner Institute: Berlin, Germany, 2009.

37. Vasilecas, O.; Laureckas, E.; Rima, A. Analysis of Using Resources in Business Process Modeling and Simulation. *Appl. Comput. Syst.* **2014**, *16*, 19–25. [CrossRef]

38. Domingos, D.; Respício, A.; Martinho, R. Using Resource Reliability in BPMN Processes. *Procedia Comput. Sci.* **2016**, *100*, 1280–1288. [CrossRef]

39. Camara, M.S.; Ducq, Y.; Dupas, R. A methodology for the evaluation of interoperability improvements in inter-enterprises collaboration based on causal performance measurement models. *Int. J. Comput. Integr. Manuf.* **2014**, *27*, 103–119. [CrossRef]

40. Saaty, R.W. The analytic hierarchy process—What it is and how it is used. *Math. Model.* **1987**, *9*, 161–176. [CrossRef]

41. Bocciarelli, P.; D'Ambrogio, A.; Giglio, A.; Paglia, E. Modeling Resources to Simulate Business Process Reliability. *ACM Trans. Model. Comput. Simul.* **2020**, *30*, 1–25. [CrossRef]

42. Smithers, T.; Posada, J.; Stork, A.; Pianciamore, M.; Ferreira, N.; Grimm, S.; Jimenez, I.; Di Marca, S.; Marcos, G.; Mauri, M.; et al. *Information Management and Knowledge Sharing in WIDE*; European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology: London, UK, 2004.