Collaborative Design and Supervision Processes Meta-Model for Rationale Capitalization

Widad Es-Soufi[1], Esma Yahia[1] and Lionel Roucoules[1*]

[1] Arts et Métiers ParisTech, CNRS, LSIS, 2 cours des Arts et Métiers 13617 Aix en Provence, France

* {Widad.ES-SOUFI, Esma.YAHIA, Lionel.ROUCOULES}@ensam.eu

Abstract   Companies act today in a collaborative way, and have to master their product design and supervision processes to remain productive and reactive to the perpetual changes in the industrial context. To achieve this, authors propose a three-layers framework. In the first layer, the design process is modelled. In the second, the traces related to the decisional process are captured. In the third, both the collected traces and the design context model are used to support decision-making. In this paper, authors address the first two issues by proposing a meta-model that allows one to capture the process' decisional knowledge. The proposal is presented and then illustrated in a case study.

1 Introduction and research background

The research reported in this paper is interested in the *product design* and *supervision* processes, a brief definition is provided of each.

The *product design* is a process in which an output (i.e. product) of a high added value is produced. It consists of modelling activities that use different resources in order to transform an input into an output that respects the imposed constraints. The *product design* also consists of decisional activities that aim at choosing one or several solutions, among all the design alternatives, based on some performance criteria. The *product design* is a complex decision-making process. Indeed, the decisions are made by several actors and have a major impact on the final product. In [1], authors have shown that 85% of the decisions, that were made in this phase, impact more than 80% of the product final cost.

The *supervision* is a decisional activity carried out by a supervisor to survey and control the progress of an industrial process. It is a decisional activity that generates an action depending on both the supervision result and the set-point. The

supervision is also a complex decision-making process for two reasons. First, the supervisor that surveys an industrial process should make, in the shortest time, the right decision in case an alarm is received. Second, the decision that is made has an impact on the supervised industrial process.

In order to master these complex processes, authors propose a three-layers framework [2]. The first layer uses a process meta-model, that captures the knowledge of the design and supervision processes, in order to model them and thus it helps companies to understand them. The second layer uses a trace meta-model to capture the design and supervision rationale and thus facilitates the decisions retrieval which is one of the main time loss reasons. The third layer analyses the captured knowledge and proposes the most suitable design or supervision process to be followed according to the industrial context. In this paper, authors address the first two layers by proposing a meta-model that models and captures the design and supervision knowledge through the 6W's concepts traceability [3].

The remainder of this paper is organized as follows. In section 2 related work is presented. In section 3, related work with respect to the aspects bounding our research context is discussed. In section 4, the proposed meta-model is presented and added values are discussed. In section 5, the proposal is illustrated in a design example. Section 6 presents the future work and concludes the paper.

2 Related work

Companies are recognizing that process modelling is a higher priority as there is an increasing need to master, understand and improve their processes. In the context of collaborative engineering, a multitude of research was interested in process meta-modelling. In [4], authors introduce the PPO model (Process, Product, Organization) which is partly based on the GRAI[1] methodology. PPO describes the relation between the triplet: Product data, Processes in which data transit and Organizations where these processes run. In [5], authors focus their research on process modelling and knowledge traceability to manage conflicts. In [6], authors establish a conceptual data model to evaluate and track design decisions in a mechatronic environment. In [7], authors identify knowledge constructs for design rationale in order to manage changes. In [8], authors propose a FBS (Function, Behaviour and Structure) based model that allows one to model the enterprise objects according to four views: process view, product view, resource view and the external effect view. The meta-models of some modelling languages, such as BPMN[2], UML[3] and IDEF0[4], also capture some of the process knowledge.

---

[1] https://en.wikipedia.org/wiki/GRAI_method

[2] http://www.bpmn.org/

[3] https://en.wikipedia.org/wiki/Unified_Modeling_Language

[4] https://en.wikipedia.org/wiki/IDEF0

3 Discussion of related work

In this section, the studied meta-models are compared according to the following three points of view bounding our context. First, the **modelling capability** that is the most important point of view; it concerns the ability of the meta-model to express the knowledge that we want to capture. The six considered criteria are the 6W's concept themselves that are described in our context as follows:

- *Who*: it is the ability to model the actor that performs the activity; namely its name, role, skills, etc. The actor is considered as a human resource.
- *What*: it is the ability to describe the product data (i.e. the input and output data) needed to execute the activity. In the context of product design, this criterion refers to both the input and the output solution spaces. Whereas, in the supervision context, this criterion refers to the state of the supervised industrial process before and after making the decision.
- *When*: it is the planned and real start time as well as the planned and real end time of the execution of the activity.
- How: it is the set of resources (material, software, human, etc.) used to execute the activity.
- *Where*: it refers to the activity in question, among the process activities.
- *Why*: it is the justification of all the choices that were made during the execution of the activity.

Second, the **representative point of view**. It concerns the external view of the meta-model and describes its ability to be both simple and well expressed. Authors define five criteria as follows:

- *Simplicity*: it describes the meta-model's level of complexity. A simple meta-model is more practical since it is easily understood and efficiently alterable if any change is detected in the organization. Simplicity can be characterized by the number of concepts describing the meta-model, as well as the quality of their graphical signification [9].
- *Richness*: it describes the ability of the meta-model to represent the knowledge inside the organization. It refers to the number of concepts and their power of expression [9]. A meta-model is literally rich if it is able to be expanded.
- *Norm*: it introduces the syntax and the semantics characterizing the grammar and the mathematical meaning of the meta-model's concepts, respectively. A normed meta-model is easily understood and verified.
- *Notation*: it describes how the meta-model's concepts are represented (graphically, textually, in the form of mathematical equations, etc.)
- *Software support*: it describes whether a tool supporting the meta-model exists.

Third, the **methodological point of view**. This aspect concerns the systematic approach of the meta-model. Authors identify three criteria as follows:

- *Granularity*: it is the process's level of abstraction also called decomposition. We need a meta-model that permits a full architectural description, i.e. the total decomposition of process into a set of sub processes and activities.
- *Consistence*: it means that both the meta-model and all its concepts should make sense. Redundant or irrational concepts have to be eliminated. This criterion is defined in our context as the capacity of the meta-model to describe a specific problem by including the needed concepts without preventing it to be expanded and thus rich.
- *Instantiation*: this criterion is defined as the implementation level of the model in order to assess whether a software supports the instantiation of the meta-model.

All the meta-models, presented in Section 2, have a norm and allow one to graphically express their concepts. The PPO meta-model ([4]) has a fairly good modelling capability since it completely models the *Who, How*, and *Where* concepts. In addition, it is fairly simple and rich, consistent and allows us a total granularity. The meta-model of Ouertani et al. ([5]) has a good modelling capability since it completely models the *Who*, *When*, *How*, *Where* and *Why* concepts. It is simple, rich, consistent, instantiable and allows us a total granularity. The meta- model of Couturier et al. ([6]) has a limited modelling capability since it models just the *Who* concept. It is fairly simple and rich, inconsistent, instantiable and does not allow us a total granularity. The meta-model of Moones et al. ([7]) has a very good modelling capability since it models all the 6Ws concepts besides being simple, rich, consistent, instantiable and allowing us a total granularity. The FBS-PPRE meta-model ([8]) has a limited modelling capability since it allows us to model just the *What* and *Where* concepts. It is not simple but fairly rich, consistent and allows one a total granularity. The BPMN and UML meta-models have a fairly good modelling capability since they model the *Who*, *What*, *How* and *Where* concepts. They are rich and fairly consistent. However, they are not simple. The IDEF0 meta-model has a good modelling capability since it completely models the *Who*, *What*, *How*, *Where* concepts and partially models the *Why* concept. However, it doesn't allow a total granularity.

The studied meta-models do not meet the totality of our requirements since they were proposed under different contexts. It is, therefore, necessary to extend some of them to construct a meta-model that perfectly matches our requirements. Authors choose to extend the IDEF0 and BPMN meta-models by specifying their concepts (for example, the IDEF0 resources are extended to human, hardware, software and documentary resources and the BPMN input is extended to input, constraints and resources). Authors also extend the meta models identified in [4], [5] and [7] since they model much of the 6Ws concepts besides being simple, rich, consistent and allowing us to express the total granularity of a process.

4 Proposal overview

The meta-model presented in Fig. 1 is the proposal of this research. It captures the design and supervision knowledge. Namely, the decisions that were taken and the choices that were rejected, while supervising a process or designing a product.

The different use cases, that may be encountered, when creating a process within the context of collaborative design and supervision are identified. First, the user starts by creating a process (cf. Process class in Fig. 1) and providing the related information including the name and the objective of the process as well as the name of the user that is creating it. Second, the user creates the different activities (cf. Activity class) that may be either modelling, decision or supervision activities. The user describes the activity by providing its name, description, type (i.e. modelling, decision or supervision), state (i.e. available to be executed, in progress or validated), real start and end time, event (i.e. *start* if the activity is the first to be executed, *end* if it concludes the process or *Null* otherwise) and the successor gateway which refers to the nature of the link between the current activity and the one that will follow [10, Sec. 8.3.9].

An activity can be either planned by the engineer (cf. PlannedActivity class) or unplanned i.e. not defined in the process model (cf. UnplannedActivity class). Indeed, sometimes during the execution of the process, some unplanned activities need to be performed when an opportunity or an obstacle comes along. For example, it is impossible to execute the machining process if there are no enough raw materials. The unplanned task here is to execute the supply activity. If the activity is already planned, the user should identify both the time in which the execution is supposed to start and the time in which it is supposed to end. Otherwise, if the activity is unplanned, he should explain the reason behind its occurrence.

An activity may have an input and should produce an output, both of them are called product data (cf. ActivityInputOutput class). We assume that the objective of this paper is mainly to retrieve the product data no matter how they are structured. Indeed, we propose to store the input and output data in a product database in a way that they can, at any time, be accessible and exploited by the running process. In the case where the activity is re-executed, the stored product data file will be incremented automatically and saved in the product database.

During its execution, an activity is supported by human, software, documentary and/or hardware resources (cf. Resource class). The user describes the context of each used resource. For example, the machine that is used during the execution of an activity must be well described in terms of its availability and trust factor. This latter is important to have some understanding on the well-functioning of the machine. An activity is constrained by some controls (cf. ActivityControl class). They could be internal (cf. InternalControl class) like the constraints imposed from anterior activities that belong to the same process. Controls could also be external (cf. ExternalControl class) like the specification imposed by the customer or the set-point related to the supervision activity. Another type of controls concerns

the decision activity (cf. DecisionActivityControl class), it is based on the performance indicator characterized by its name, type and priority.
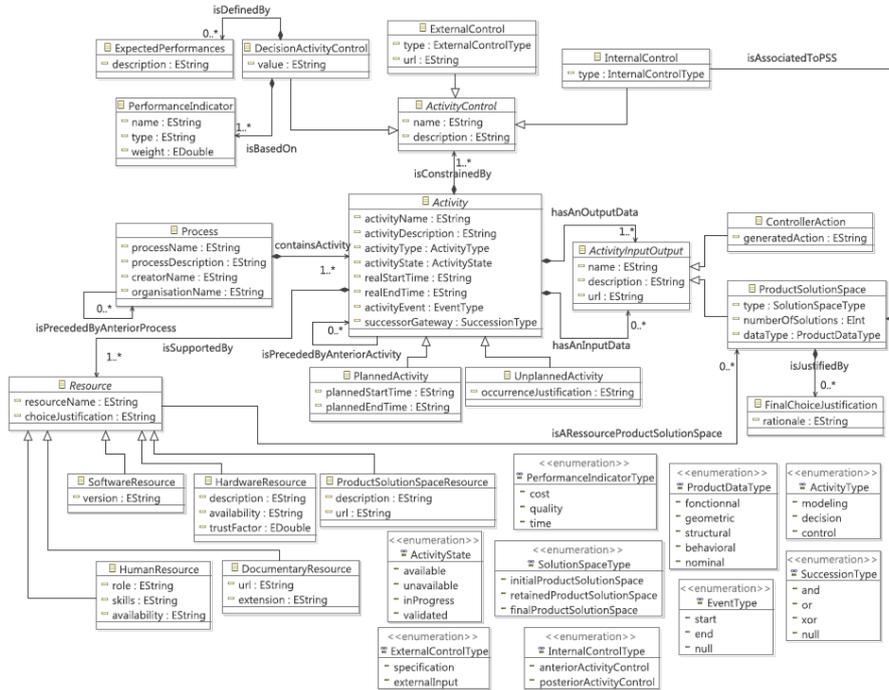

Fig. 1. The proposed meta-model for modelling and tracing the design and supervision processes

The proposed meta-model is implemented in Eclipse[5] and allows one to model and trace the design and supervision knowledge. Indeed, authors assume that it is important to trace all the knowledge constructs identified in Fig. 1. Therefore, the proposed meta-model is instantiated in Eclipse to create real world models and generate a XMI (XML Metadata Interchange) trace, that can be stored in a process trace base. Authors assume that the proposal allows companies to understand their design and supervision processes through the process modelling. They also assume that, throughout the knowledge traceability, the proposal helps companies to gain the time that they usually lose when retrieving the decisional information.

5 Case study: collaborative design of an electric torch

The considered design process contains eleven interdependent activities and involve many engineers working together to design an electric torch. Engineers are

---

[5] https://eclipse.org/

asked to: (1) Describe how the electric torch may be used by highlighting its functions. (2) Study in-depth the product functions which are realized through a physical principle by a specific technology. (3) Describe for each function its energetic properties. (4) Provide an approach to find technology solutions related to the functions. (5) Identify and describe the products used in the design. Finally, (6) Give a first CAD model of the product and progressively refine it.

The proposed meta-model (Fig. 1) is instantiated to create the electric torch design trace (Fig. 2). This latter captures all the design knowledge including the process context, the process activities (*Where*), the engineers that were performing these activities (*Who*), the date when they performed them (*When*), the rationale behind their choices (*Why*), the resources that they used to execute these activities (*How*), and the results of the execution of these activities (*What*).

```
processName="Electric Torch Design Process"
processDescription="This process shows the activities that should be performed to de-
sign an electric torch"
     creatorName="Lionel Roucoules"
     organisationName="ENSAM">
  <containsActivity
     xsi:type="PP:PlannedActivity"                                    Where
     activityName="A1: External functional analysis "
     activityDescription="The activity will analyze, describe and evaluate the functions
that the lamp must execute"
         realStartTime="01-08-2015"
         realEndTime="01-08-2015"
         plannedStartTime="01-08-2015"        When
         plannedEndTime="01-08-2015">
     <isSupportedBy
        xsi:type="PP:HumanResource"
        resourceName="Rafael Dawson"
        choiceJustification="Expert in functional analysis"     Why
        role="Industrial engineer"
        skills="TDC Need, product development"
        availability="Yes"/>
     <isSupportedBy
        xsi:type="PP:HumanResource"
        resourceName="Tom Stilinski"                                          Who
        choiceJustification="Expert in functional analysis"     Why
        role="Industrial engineer"
        skills="TDC Need"
        availability="Yes"/>
     <isSupportedBy
        xsi:type="PP:HumanResource"
        resourceName="Lydia Stanford"
        choiceJustification="Expert in functional analysis"0    Why
        role="Industrial engineer"
        skills="TDC Need"
        availability="Yes"/>
     <isSupportedBy
        xsi:type="PP:SoftwareResource"
        resourceName="TDC Need"                                  How
        choiceJustification="It is very simple to use"    Why
        version="2015"/>
     <hasAnOutputData                                                   What
        xsi:type="PP:ProductSolutionSpace"
        name="APTE diagram"
        description="This method focuses on the functional analysis at the early stages
of the design process"
        url="C:\ElectricTorchDesign\APTEDiagram"
        type="finalProductSolutionSpace"
        numberOfSolutions="1"
        dataType="behavioral"
        kind="output"/>
  </containsActivity>
</PP:Process>
```

Fig. 2. Part of the generated XMI trace

6 Conclusion

This paper proposes a collaborative design process meta-model whose objective is to model and trace the design and supervision rationale. This helps companies to manage their processes to be more productive and reactive to changes. Indeed, the proposed meta-model helps structuring the enterprise's processes which makes easy their understanding. It also helps documenting the decisional process and memorizing the rejected choices. Future work consists in learning from the process traces, that were generated by the proposed meta-model, to support engineers in their decisions-making processes.

References

[1]   C. Berliner and J. A. Brimson, *Cost Management for Today's Advanced Manufacturing: The CAM-I Conceptual Design*. Harvard Business School Press, 1988.

[2]   L. Roucoules, E. Yahia, W. Es-Soufi, and S. Tichkiewitch, "Engineering design memory for design rationale and change management toward innovation," *CIRP Annals - Manufacturing Technology*, 2016.

[3]   J. A. Zachman, "A Framework for Information Systems Architecture," *IBM Syst. J.*, vol. 26, no. 3, pp. 276–292, Sep. 1987.

[4]   P. Nowak, B. Rose, L. Saint-Marc, M. Callot, B. Eynard, L. Gzara-Yesilbas, and M. Lombard, "Towards a design process model enabling the integration of product, process and organization," in *5th International Conference on Integrated Design and Manufacturing in Mechanical Engineering, IDMME*, 2004, pp. 5–7.

[5]   M. Ouertani, L. Gzara-Yesilbas, and G. Ris, "A Process Traceability Methodology to Support Conflict Management," in *Proceedings of the 10th International Conference on CSCW in Design, CSCWD 2006, May 3-5, 2006*, pp. 471–476.

[6]   P. Couturier, M. Lô, A. Imoussaten, V. Chapurlat, and J. Montmain, "Tracking the consequences of design decisions in mechatronic Systems Engineering," *Mechatronics*, vol. 24, no. 7, pp. 763 – 774, 2014.

[7]   E. Moones, E. Yahia, and L. Roucoules, "Design process and trace modelling for design rationale capture," in *Joint Conference on Mechanical, Design Engineering & Advanced Manufacturing*, 2014.

[8]   M. Labrousse and A. Bernard, "FBS-PPRE, an enterprise knowledge lifecycle model," in *Methods and tools for effective knowledge life-cycle-management*, Springer, 2008, pp. 285–305.

[9]   F. Daoudi and S. Nurcan, "A benchmarking framework for methods to design flexible business processes," *Software Process: Improvement and Practice*, vol. 12, no. 1, pp. 51–63, 2007.

[10]   O. M. G., "Business Process Model and Notation (BPMN) Version 2.0," Jan. 2011.