

Knowledge and Information Structuring in Reverse Engineering of Mechanical Systems

Mohamed Islem Ouamer-Ali^{1(✉)}, Florent Laroche¹,
Sébastien Remy², and Alain Bernard¹

¹ IRCCyN, Ecole Centrale de Nantes, Nantes, France
{m.ouamerali, florent.laroche,
alain.bernard}@irc cyn.ec-nantes.fr

² ICD, Université de Technologie de Troyes, Troyes, France
sebastien.remy@utt.fr

Abstract. In an increasingly competitive environment in the manufacturing industry, the control of time, cost and performance enables companies to stand out and take the lead. We are witnessing the proliferation of design aiding solutions that support the designer in his work, and it is through these solutions that developers and researchers aim to improve product development, and control all the project management aspects raised above. One of the activities that are used in product development is Reverse Engineering. This activity allows the extraction of information from an existing physical product. In industry, we may use this activity in order to maintain long-life products, or make a re-design, re-engineering, re-manufacturing, etc. In this paper, we propose an approach that will allow the management of a global reverse engineering process for complex mechanical assemblies.

Keywords: Reverse engineering · Product models · Knowledge management · Knowledge representation · PLM

1 Introduction

In order to introduce a comprehensive and robust methodological approach of reverse engineering, a first exploration of different methodologies was made. [1] We find in literature, different definitions and methods of reverse engineering, however, the dependence of these methods to contexts of use, and the lack of knowledge use (other than topological and geometrical) makes methods and tools local (regarding their use), and non-robust. We are aiming to fill these gaps through the integration of knowledge of different types, and the use of heterogeneous data that can be found on the analysed products.

It is therefore normal that one of the main issues is knowledge management globally, from its capture to its use in the process. In this article, we will focus more on the aspect of knowledge and information structuring, implementation and use. The structuring part will concern the implementation of a knowledge base for the identification of different products. This knowledge base will be based on an adequate product model that will allow us to take into account all knowledge and information in

a product. As for the use, it will concern the final reconstitution of the product which is not going to be explored in this article.

At first, a presentation of the state of the art required to build our approach will be made in Sect. 2. Then, the definition of reverse engineering in our perspective and what will help us develop our approach to offer a global methodology will be discussed in Sect. 3. Finally we conclude the results and future work.

2 State of the Art

2.1 Reverse Engineering

Definitions. Reverse engineering was developed as an alternative to set or reset objects or products [2]. It is the reverse of the design process. It consists of the rebuilt of a design model based on an actual product. [3] The main goal of reverse engineering is going back to the results of the original design process of a part or an assembly in order to reuse this information, as shown in Fig. 1.

According to Chikofsky [4] the Reverse Engineering process can also be defined as a process for analysing a system to:

- Identify system components and their relationships.
- Create representations of the system in another form or another level of abstraction.

Phases of Reverse Engineering. Reverse engineering has a strong presence in mechanical design. There are several solutions that have been developed in this area. However, these solutions are often context dependent, partial and incomplete [5, 6].

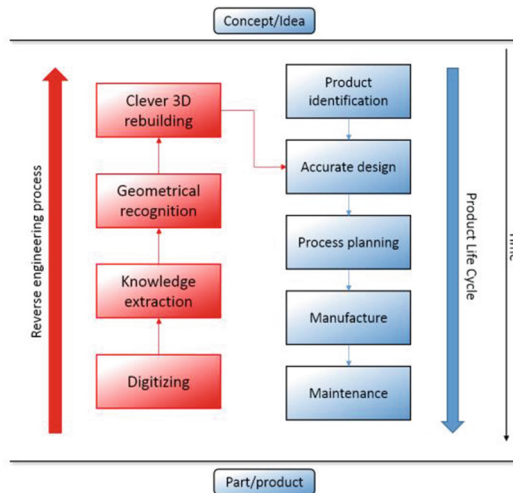


Fig. 1. The link between product lifecycle and the reverse engineering process [7].

In general in the literature, four main actions are identified in the reverse engineering process and which are:

1. Scan Product and Data Acquisition
2. Segmentation of the data acquired during the scan
3. Knowledge Extraction (i.e. Manufacturing features recognition– geometric and topologic knowledge in most cases).
4. Reconstruction of the 3D model updated

It is worthy to note that knowledge used in these solutions (if so) is restricted to only topological and geometric shapes, and therefore, these reverse engineering methods could restrict the scope of the designer, whereas other types of knowledge [7] integrated in the process could have made it semantically richer allowing the user to have more freedom in it.

2.2 Knowledge

Definitions. In order to define knowledge, several authors propose to highlight the differences between data, information, and knowledge.

Data is gross figures and facts, while information is processed data. Finally, knowledge is authenticated and contextualized information [8]. Knowledge exists outside of an intelligent system only as information (Knowledge is personalized information on facts, procedures, concepts, interpretations, ideas, observations, and judgments [8]), since they are the result of a cognitive process that leads to the binding of different pieces of information between them by means of semantic links, creating a global information schema.

Still according to [8], if we were to establish a hierarchy among these three concepts, two approaches are identified:

1. The first is where the existence of knowledge is conditioned by the existence of information, which itself is conditioned by the existence of data, so: data (low level) and information (Average level) and knowledge (high level).
2. The second is where knowledge allows us to formulate the information, and therefore to measure the data.

What one can declare, is that the three concepts are related and condition the existence of each other where without data there is no information (and therefore no knowledge), and without knowledge there is no sense to data.

Knowledge Typology. In order to address knowledge in the best way and represent it, it is necessary to make a classification by type of knowledge, and thus define the domain of definition of each type of knowledge.

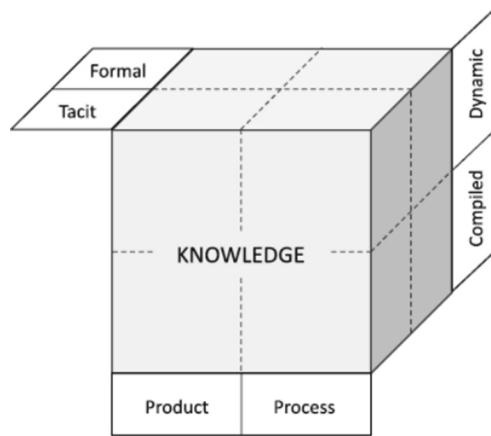


Fig. 2. Knowledge Typology by Nonaka [10].

As shown in Fig. 2, knowledge types can be classified in three different dimensions. Therefore, each classification has its own axis [10]:

- 1. Formal/Tacit:** Formal knowledge is integrated in the documents of the product, the structure and functions description of the product, etc. And tacit knowledge is knowledge related to experience, implicit rules, intuition, etc. [10].
- 2. Product/Process:** Product knowledge takes into account the information and knowledge about the evolution of the product throughout its life cycle. The process knowledge can be classified as follows: knowledge of the design process, project knowledge, and knowledge of the manufacturing process.
- 3. Compiled/Dynamic:** The compiled knowledge is mainly obtained from the experience that can be compiled into rules, plans, scripts, etc. The solutions are explicit. Dynamic knowledge uses knowledge that can generate additional knowledge structures, which are not taken into account by the compiled knowledge.

This typology will allow us to define which knowledge could be captured in order to be exchanged in a community. For now, knowledge that could be shared has to be formal since it is simpler to formalize, compiled since it concerns knowledge structure in the knowledge base, and treats products throughout their lifecycles. Formal knowledge should be shared through a channel which allows its full and comprehensive representation this is why we need to represent it in a proper way.

Knowledge Representation. Collaborative engineering implies sharing knowledge between actors from different working groups of the same project. Sharing this knowledge follows mechanisms where knowledge should be made explicit (formal). Hence, the need to represent knowledge in order to facilitate communication between actors.

In [11] knowledge representation is described as having 5 roles:

1. Substitution of an external entity (real world) which is carried out by an internal process.
2. A set of ontological commitments.
3. A fragmentary theory (partial view) of an intelligent reasoning.
4. An effective way of reasoning (artificial intelligence).
5. A means of human expression (communication).

It is also important to note that the existence of different sets of knowledge implies a difference in their representation using different modes and tools.

Knowledge Representation Modes. Knowledge representation can be classified. In [12], are proposed 5 categories of knowledge representation: pictorial, symbolic, linguistic, virtual, and algorithmic.

In [9] Based on the above categories of representation, we propose a scheme of knowledge representation in product design.

In Fig. 3 different tools of product design used in the different phases of the design process are matched with knowledge representation modes. This match is more detailed in Fig. 4.

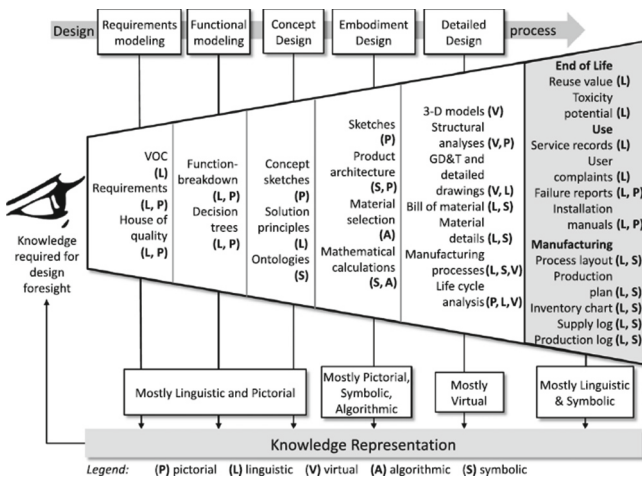


Fig. 3. Knowledge representation modes in product lifecycle [9].

Pictorial	Symbolic	Linguistic	Virtual	Algorithmic
Sketches	Decision tables	Customer Requirements	CAD Models	Mathematical Equations
Detailed drawings	Production rules	Design Rules, constraints	CAE Simulations	Parametrizations
Charts	Flow charts	Analogies	Virtual Reality simulations	Constraint Solvers
Photographs	FMEA diagram	Customer feedback	Virtual prototypes	Computer Algorithms
CAD model views	Assembly tree	Verbal communication	Animations	Design/ operational procedures
	Fishbone diagrams		Multimedia	
	Ontologies			

Fig. 4. Design tools according to knowledge representation modes [9].

The implementation of an approach based on knowledge management requires a good comprehension of the concepts. Thus, defining the right knowledge typology with the right knowledge representation modes would put a basis for structuring knowledge through a complete product model taking into account all information and knowledge covered in the case of reverse engineering, and maybe conventional design.

2.3 Product Models

A product model is a model that is used to structure the product of a specific vision. There are several product models in scientific literature.

For exemplification purposes, the FBS model shown in Fig. 5 is chosen.

The FBS model

This model categorizes any object in 3 aspects: [13]

- Function: Or, What is the purpose?
- Behaviour: Or, as the subject?
- Structure: Or, what is the object?

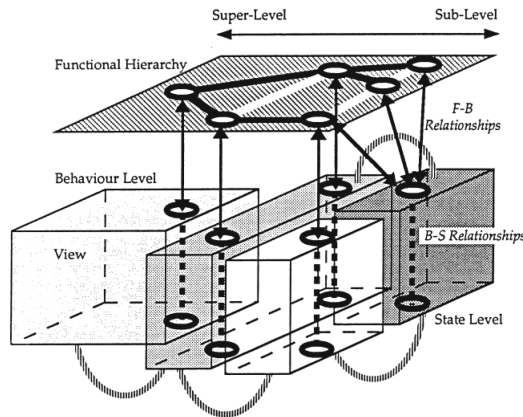


Fig. 5. A view of the FBS product model [13].

The definitions presented by Gero in this model are not clear. We note the absence of a stable definition of the function, and the double objective description of the current design and prescription of improved design [14].

The contribution with a product model in our approach consists on a global organization of information to facilitate its use, and the establishment of a solid foundation on which is based our work for a global reverse engineering methodology. This is one of the most important parts in the elaboration of our approach, and the product model that will be implemented will depend on the types of knowledge and information taken into account and their representation modes.

3 New Approach of Reverse Engineering

Considering the definition of Chikofsky [4], we can give the following proposition: “Reverse engineering is a process of moving from one level of abstraction to another,” from a low level to a higher one, and therefore, we can move from the concrete to the abstract product produced and have the results of the design. This definition is consistent with the fact that reverse engineering is the reverse process of design. Indeed, the design allows us to start from the idea (most abstract form of the product) to the product itself (the most concrete form of the product). We can see in Fig. 6 the qualitative representation of the correlation between levels of abstraction and product development cycle. The green arrow represents the reverse engineering process. We could also put an arrow in the opposite direction relative to the ‘normal’ design cycle.

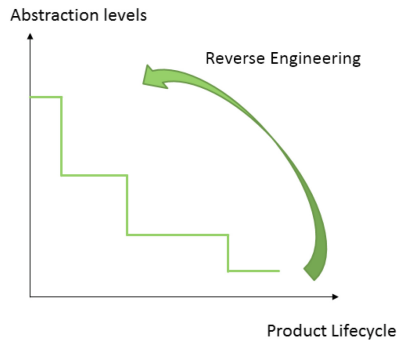


Fig. 6. Relationship between product lifecycle and abstraction levels [13].

It would be interesting to consider the abstraction of information as an important aspect in the implementation of a reverse engineering process to track its status.

3.1 Information Abstraction Levels

This concept has recently been introduced in the field of mechanical engineering in general, and has never been used in reverse engineering. It is in the field of IT that we tried to identify the levels of abstractions of information: we find for example in [15] the following levels which follow a hierarchy from most concrete to most abstract:

- **Objective Level:** represents the context in which the product is placed and its objective. For example for a gear, the objective is the transmission of motion between two coaxial shafts.
- **Conceptual Level:** represents a first definition of the product through information deduced from its objective (goal). For example, the input and output parameters in the process of motion transmission.

- **Functional Level:** represents the functions that would be accomplished by the product. For example, the gear transmits power through its cogs, follows the movement of the shaft through a spline, etc.
- **Logical Level:** represents information related to the product's behaviour. For example, the relationship between the geometry of a cog in a gear, the material used and the torque limit.
- **Physical Level:** represents the most concrete level with topological and geometrical information. For example the shape of a gear's cogs in a CAD model.

Based on the levels defined above, we can identify a first correspondence between the abstraction levels and different aspects of product models. For example there are the aspects of the FBS model in the last 3 levels of abstraction: functional level (Function in FBS), logical level (Behaviour in FBS), and Physical level (Structure in FBS).

However, we find the limit of the FBS model that does not take into account other levels of abstraction (Such as objective and conceptual levels) that are just as important in defining a product.

Indeed, the objective level takes into account the context in which emerged the idea of the product, and in which is defined the objective to which it should respond. And the conceptual level sets up a first segmentation of the product that will determine the rest of the product development.

3.2 Representation Modes

The inclusion of representation modes in reverse engineering allows to not only to communicate information related to knowledge represented differently, but also provides a classification of different knowledge according to their representation.

One could imagine the combination of knowledge representation with the levels of abstraction to define the state of knowledge at a certain point in time 'T' of the product lifecycle. So, this would define a space in which each point represents a state of knowledge with as characteristics, level of abstraction and representation mode. This would allow the establishment of a mean for a global methodological approach to reverse engineering, which would take into account all possible aspects of knowledge related to the product by implementing a product model that corresponds to the levels of abstraction.

3.3 Knowledge State

Knowledge state represents a set of knowledge and information at a certain point in time (represented by the product lifecycle), and a specific level of abstraction. One could imagine as in Fig. 7 where the points P, Q and R represent different states of knowledge on a product, where P is a representation of the product technical specifications, Q is its behaviour and R its functions.

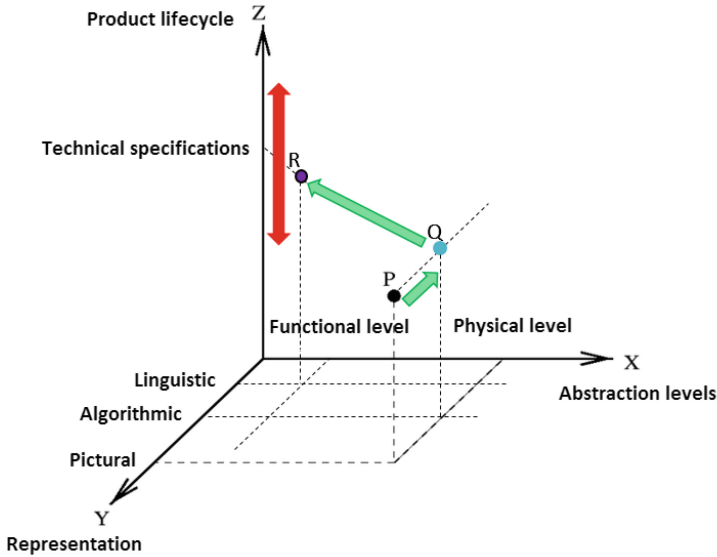


Fig. 7. Knowledge states in the states space represented by abstraction levels, representation

Reverse engineering by definition is the transition from an abstraction level to another, so we can say that this is a change of knowledge state, which would mean that reverse engineering is a process which allows us to move from one point to another in knowledge state as shown in Fig. 7.

Finally, the space in which the knowledge states are located could be used to map all the activities that contribute in the reverse engineering process.

4 Conclusion and Future Work

To establish a comprehensive methodology of reverse engineering, a new definition was proposed: reverse engineering is the process of knowledge state transformation that allows us to move from a state ‘A’ where knowledge is represented in a representation mode and level of abstraction at a certain point of time ‘T’, to a state B where one or more of the three aspects of the status changes (i.e., the time, the level abstraction, and the mode of representation).

A clarification should be made regarding the different states of knowledge and the various state changes to be considered in the reverse engineering process.

Concerning the possible implementation of this new approach, an adequate product model should be proposed in order to take into account all aspects of information and knowledge required for reverse engineering process. It will also identify the correspondence between the different types of knowledge initially defined, and the state of knowledge.

References

1. Ouamer-Ali, M.I., Laroche, F., Bernard, A., Remy, S.: Toward a methodological knowledge based approach for partial automation of reverse engineering. *Procedia CIRP* **21**, 270–275 (2014)
2. Bernard, A.: Reverse engineering for rapid product development: a state of the art. In: *Proceedings of SPIE- The International Society for Optical Engineering, Three dimensional* (1999)
3. Benko, P., Varad, T.: Segmentation methods for smooth point regions of conventional engineering objects. *Comput. Aided Des.* **52**, 36–511 (2004)
4. Chikofsky, E.J., CROSS, J.H., et al.: Reverse engineering and design recovery: a taxonomy. *IEEE Softw.* **7**(1), 13–17 (1990)
5. Remy, S., Ris, G., Nartz, O., Bernard, A.: Reverse engineering of a 1935 Delahaye radiator cap. In: *European Virtual Engineering Network Conference 2003, Dublin, Irlande*, pp. 107–115, ISBN 0-947974-15-6 (2003)
6. Bernard, A., Nartz, O., Remy, S., Ris, G., Zhang, Y.F., Loh, H.T., Wong, Y.S.: Reverse engineering and rapid prototyping. In: *Software solutions for Rapid Prototyping*, pp. 283–339, Chap. 9, publishing Ed., UK, Coordinator: Ian Gibson, ISBN: 1-86058-360-1 (2002)
7. Troussier, N., Bricogne, M., Durupt, A., Belkadi, F., Ducellier, G.: A knowledge-based reverse engineering process for CAD models management. In: *Proceedings of IDMME-Virtual Concept* (2010)
8. Alavi, M., Leidner, D.E.: Review: knowledge management and knowledge management systems: Conceptual foundations and research issues. *MIS Q.* **25**(1), 107–136 (2001)
9. Chandrasegaran, S.K., Ramani, K., Sriram, R.D., et al.: The evolution, challenges, and future of knowledge representation in product design systems. *Comput. Aided Des.* **45**(2), 204–228 (2013)
10. Nonaka, I., Konno, N.: The concept of “ba”: building a foundation for knowledge creation. *Calif. Manage. Rev.* **40**(3), 40–54 (1998)
11. Davis, R., Schrobe, H., Szolovits, P.: What is a knowledge representation? *AI Mag.* **14**(1), 17–33 (1993)
12. Owen, R., Horváth, I.: Towards product-related knowledge asset warehousing in enterprises. In: *Proceedings of the 4th International Symposium on Tools and Methods of Competitive Engineering (TMCE 2002)*, pp. 155–70 (2002)
13. Gero, J.S., Rosenman, M.A.: A conceptual framework for knowledge-based design research at sydney university’s design computing unit. *Artif. Intell. Eng.* **5**(2), 65–77 (1990)
14. Vermaas, P.E., Dorst, K.: On the conceptual framework of John Gero’s FBS-model and the prescriptive aims of design methodology. *Des. Stud.* **28**(2), 133–157 (2007)
15. Rasmussen, J.: The role of hierarchical knowledge representation in decision making and system management. *IEEE Trans. Syst. Man Cybern.* **2**, 234–243 (1985)