Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers ParisTech researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: https://sam.ensam.eu
Handle ID: .http://hdl.handle.net/10985/11400

To cite this version :

Samira SADEGHI, Thomas DARGON, Louis RIVEST, Jean-Philippe PERNOT - Capturing and analysing how designers use CAD software - In: Tools and Methods for Competitive Engineering (TMCE'16), France, 2016 - Proceeding of Tools and Methods for Competitive Engineering - 2016

Any correspondence concerning this service should be sent to the repository Administrator : archiveouverte@ensam.eu

CAPTURING AND ANALYZING HOW DESIGNERS USE CAD SOFTWARE

Samira Sadeghi
Arts et Métiers ParisTech
Laboratory LSiS
sadeghisamirafr@gmail.com

Thomas Dargon
Arts et Métiers ParisTech
Laboratory LSiS
thomas.dargon@gadz.org

Louis Rivest
École de technologie supérieure
Département de génie de la production automatisée
louis.Rivest@etsmtl.ca

Jean-Philippe Pernot
Arts et Métiers ParisTech
Laboratory LSiS
jean-philippe.pernot@ensam.eu

ABSTRACT

Current Computer-Aided Design (CAD) packages support the storage of the final design models and solutions in different formats, and PLM software manages the high-level information about the design process, such as the versioning of the design solutions. However, the processes happening inside the CAD software are not being fully captured. Information such as the sequence of actions (create a sketch, set a distance constraint, remove a pocket, modify the diameter of a through hole, etc.), versioning of the created objects, etc. is missing. This information can be used to understand how a designer uses CAD software to generate geometric representations. In design companies, capturing this information during a product design project would help to evaluate the designer's way of working with CAD software. In design education, collecting information on how design students generate geometric representations would allow teachers to identify the areas of misunderstanding, improve the education process by representing the optimal way of working, and help teachers to correctly evaluate their students' performance in using CAD software. This paper proposes a framework to support an analysis of how designers use CAD software to generate geometric representations. This framework consists of structured models and an approach which guides the actor in capturing the design process. We use CATIA as a CAD software solution, but the proposed approach is generic and can be extended to any CAD software. The validity of the proposed approach is illustrated through a case study.

KEYWORDS

CAD, modeling, design process, modeling process, solid modeling, metrics, performance

1. INTRODUCTION

In designing a product, not only are design ability and design skills important; being skilled in using the supporting tools is also necessary for the success of a well-designed product. One type of supporting tool is CAD software. CAD tools are used in the more detailed design stages to create detailed design tasks, to create geometries in digital format in both 2D and 3D, to capitalize on the design information, to facilitate the numerical calculations, enhance the precision of the design, etc.

Defining a unique method for the creation of geometries in CAD software is not easy because of the wide variety of functions available within this tool. Different combinations of software commands can result in the same geometry (same final model). Moreover, the modeling process itself evolves and is difficult to predict. Often, the designer does not have a fixed method for designing a specific part or product. The method changes in terms of the results of each action and the difficulties encountered. These modeling processes are often long (depending on the difficulty of the part and the user's skill level). Thus, it is difficult to define a best practice or best process to follow for creating a specific geometry.

Myers et al. [1] argue that monitoring the actions taken by a designer with a CAD tool would provide a rich, semantically-grounded process history for detailed design. Currently, the process information is not associated with the CAD file, indicating that

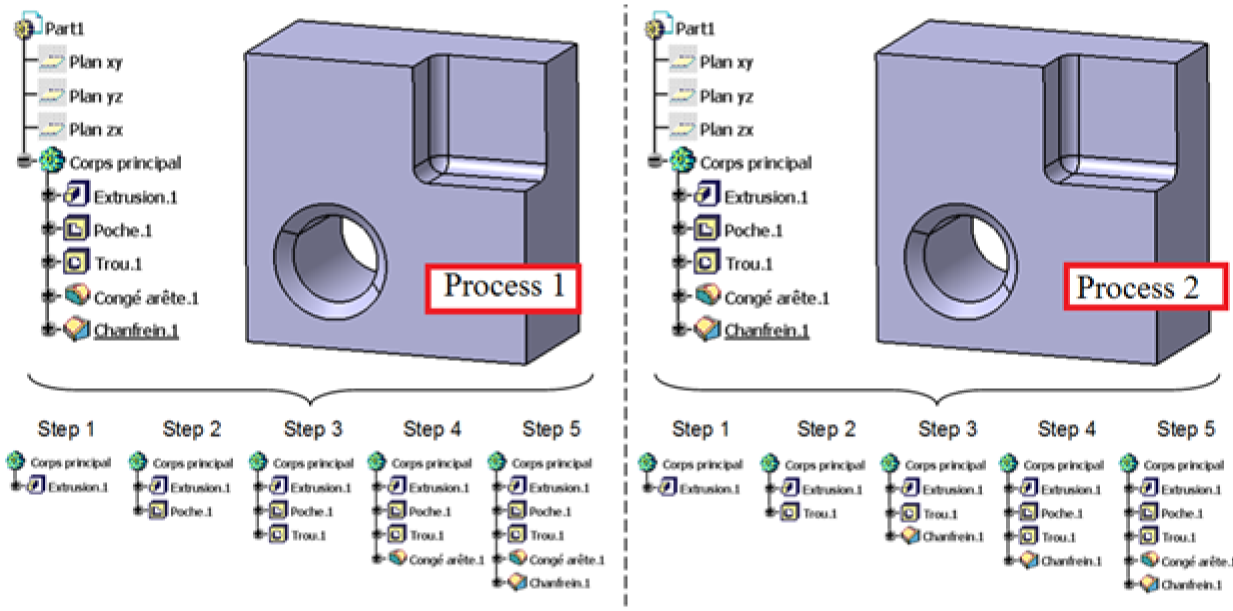

Figure 1 Two different modelling process generating the same construction trees by following different steps. Observe the resulting trees (step 5); it is impossible to distinguish the two processes.

there is no link between the geometric model and the user's modeling process.

The design of a CAD model results from successive steps of operations and manipulations partially stored in a construction tree. Therefore, by reading the construction tree from the root one can define the succession of operations which generate that CAD model. However, the construction tree does not capture the multiple and potentially iterative steps that were followed to reach the final tree, because that history is not stored within the construction tree. In the example shown in figure 1, modelling processes 1 and 2 give rise to the same construction tree, even if the successive steps followed to reach the final tree are not the same. In step 4 of process 2, the designer has inserted a pocket in between the extrusion and the hole. Similarly, in process 1, the initial length of the first extrusion has been modified several times. This information has not been captured and thus does not appear on the final trees.

In addition, the time associated with each entity stored in the construction tree is only the date of creation and last modification date of that entity, in the format "hh: mm", which does not allow for great precision in a study of action temporality during the modeling process.

The difference between the users' modeling processes is due to the difference in their CAD knowledge. Bhavnani et al. [3] discussed the results of studying the behavior patterns of different types and levels of CAD users for a specific drawing task. Their results show that the experience of the CAD user has a clear correlation to the pattern of commands used, the time taken, and the quality of the drawing produced. Chester [2] proposed three types of CAD knowledge:

- *Declarative command knowledge* corresponds to knowledge about the commands or algorithms that are available within CAD software. A user needs to know about these algorithms before he/she can use them and he/she needs to know that they exist so that he/she can find them in new software. This knowledge is generic in nature and is applicable across the majority of CAD software.
- *Specific procedural command knowledge* corresponds to knowledge that enables an operator to execute the necessary commands. Specific procedural command knowledge thus varies from one CAD software package to another, and may also vary from one version of a CAD software package to another.
- *Strategic knowledge* includes a range of Meta cognitive processes. Being aware that this type of knowledge occurs ensures that the choice of algorithms will enable the construction of a model that can actually be manufactured.

Extracting the above-mentioned knowledge types from design processes and defining the best practices requires capturing and analyzing the information about those design processes.

1.1. Motivations for capturing the design process inside CAD software

We have identified three domains where in it could be worthwhile to capture how designers use CAD software to generate geometric representations:

Universities and educational scenarios: Learning or teaching how to use CAD software poses challenges for both trainers and trainees. Capturing the student's modeling process can help a teacher to adapt their teaching according to the difficulties faced by each student. This can also help teachers to evaluate their students' work. For example, while two students may each submit a correct geometric model, one may have succeeded in their first attempt, but the other could have spent more time and followed more modeling steps. Based on the teacher's rating criteria, knowing this background information will help to distinguish the work of these two students.

Industrial scenarios: For companies that use CAD software on a daily basis, capturing design processes could be helpful in several ways. Information about these processes could help in identifying obstacles to the development of solutions, reduce modeling time and increase team efficiency. Specifically, it would help to identify good practices based on the expert's modeling methods. Capturing and analyzing employees' design processes may encourage them to reflect on their working method and to improve it. Capturing the design process and the automatic generation of models representing users' modeling processes could help companies to check whether their employees have implemented the modeling rules set by the company. These modeling rules are imposed to support robustness as well as to facilitate the exchange of models between different designers.

CAD software developers' scenarios: Users' design processes can provide feedback that CAD software developers can use to improve the software. For example, if they realize that 80% of users use two functions one after the other, then in a future release, they may offer a new command which combines these two functions, thus increasing efficiency. Similarly, if they see that many people spend a lot of time on the definition of a particular object, they could add functionalities to the software that will facilitate the development of such objects.

1.2. Literature review

Several studies have investigated design process modeling and the capture of design information. Nomaguchi & Fujita [4] proposed a knowledge representation framework, DRIFT (Design Representation Integration Framework of Three layers). The core of DRIFT is a three-layered design process model of actions, operations, and argumentation which captures and manages the reflection processes of generating and verifying design concepts. Ishino & Jin [5] proposed a three-layer design process model to represent generic design processes: 1) Event-Layer: captures primitive-level design events that are generated by designers operating a CAD system; 2) Operation-Layer: represents higher-level design operations that reflect meaningful design actions, which are clusters of plural events; and 3) Product Model-Layer: represents the design alternatives that are generated from multiple design operations. Later, in [6], they proposed a design activity knowledge acquisition (DAKA) framework that extracts designers' design activity knowledge from the CAD operation event data. Their framework is composed of a product model and a function-based design operation-mining algorithm for extracting meaningful design operations from CAD event databases. Sung et al. [7] developed a method to record how and why a design has arrived ata given stage. They captured design processes and design knowledge by logging individual designer behavior and system interactions while a CAD system is being used. Sivanathan et al. [8] propose a generic framework for the ubiquitous multimodal data capture of CAD system activities. The metadata is captured and stored using ubiquitous multimodal capture tools embedded in the design environment. This data capture is automatic and embedded in the working environment. Consequently, the designer/engineer has no interruption in their usual design activity, and there is no extra workload involved in capturing and generating the information about the design process.

Several studies have investigated the development of methods and tools to capture the design rationale during a design process [9], [10], [11], and [12]. Design rationale retains design knowledge such as design assumptions, constraints and design reasoning that are often not captured. It captures design alternatives in order to help understand why some designs have been rejected [13]. Myers et al. [14] proposed the Rationale Construction Framework (RCF) that acquires rationale information for the detailed design process. The underlying approach involves monitoring designer interactions with a commercial CAD tool to produce a process history. Bracewell et al. [17] proposed Design Rationale editor (DRed), a software tool that allows engineering designers to record their rationale as the design proceeds.

None of the works found in the literature include a specific investigation on understanding the way in which designers use CAD software to generate geometric representations.

1.3. Focus of our paper

The global objective of this research study is to offer a means to analyze how designers use CAD software to generate geometric representations. Because of the diversity of peoples' motivations for capturing the design process inside CAD software and the multiple ways designers design using CAD, we propose a framework to guide an actor (*a person who seeks to capture the design process with a specific objective*) so that they can achieve their objectives.

The proposed framework consists of structured models and an approach that guides an actor in capturing the design process.

The proposed models are introduced in section 2, and the proposed approach and corresponding steps are presented in section 3. In section 4, we introduce our specific scenario and show how we can achieve the objectives of our scenario by following the proposed approach step-by-step. We conclude in section 5, where we present the discussion and our conclusions.

2. PRPOSED MODELS FOR CAPTURING THE DESIGN PROCESS INSIDE CAD SOFTWARE

To support the actor in identifying the elements to be captured from the design process, we built a representative model and a data model of the design process. We considered several criteria for these models.

The models should:

- Represent the design process, showing how the work is actually performed;
- Support comprehensive analysis of the process through the model;
- Provide a flexible and easily understandable model for representing the process and the required level of detail based on scenario requirements;
- Accurately represent the behavioral aspects of the process and track designer operations by providing a rich set of design operations;
- Record the evolving attribute values, both structural and semantic;
- Record the relationships between the objects, the object-process and the processes; and
- Generate a stream of tool events and capture the time notion of the process.

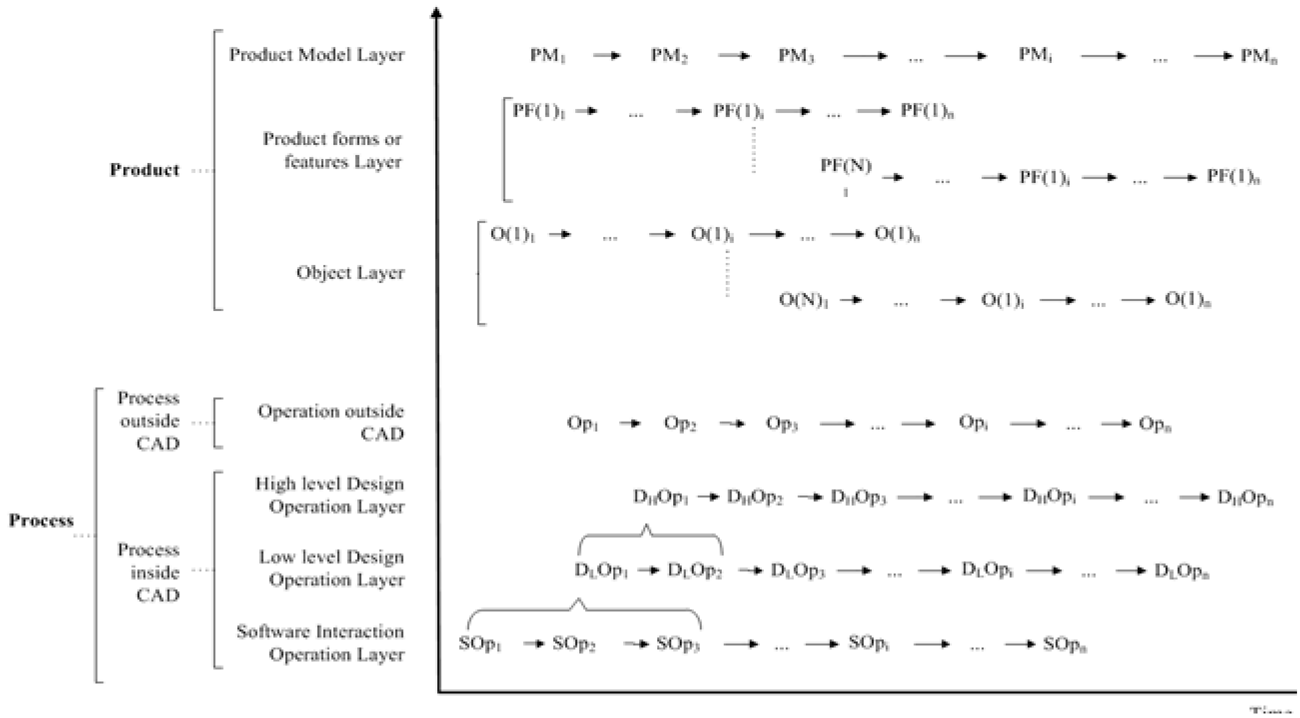

Figure 2 The multi-layered design process model

2.1. Multi-Layered Design Process Model:

The first model we developed is a Multi-layered design process model that provides a representative model (Figure 2). This model is based on two main information layers: *Product* and *Process*. The Product Layer consists of three main layers: An *Object Layer, Product forms or features Layer,* and a *Product Model Layer.* The Process Layer has two main layers: The *Process inside the CAD* and the *Process outside the CAD*. The process inside the CAD contains three layers: A *Software interaction operation layer,* a *Low level design operation layer,* and a *High level design operation layer.*

In the product layers, the '**i**' in '$\mathbf{PM_i}$' is the index of the state of the product model. The '**i**' in '$\mathbf{O(j)_i}$' is the index of the object state and the '**j**' is the identity of the object. '$\mathbf{PF(j)_i}$' is the specific type of the object, which is more meaningful in terms of design and is usually the aggregation of several objects. The '**i**' in '$\mathbf{PF(j)_i}$' is the index of the state and the '**j**' is the identity of the product form/feature. A '**PMi**' is the aggregation of several '$\mathbf{O(j)_i}$' s.

'$\mathbf{O(j)_i}$' is created or modified by a sequence of high level or low level operations (e.g. $\mathbf{D_H O_{Pi}}$ and $\mathbf{D_L O_{Pi}}$). High level operation $\mathbf{D_H O_{Pi}}$ is performed through a sequence of low level operations (e.g. $\mathbf{D_L O_{Pi}}$). $\mathbf{D_L O_{Pi}}$ is performed through a sequence of software interaction operations (e.g. $\mathbf{SO_{Pi}}$: selecting the extrude function from the menu/shortcut to trigger the extrude function) (See Figure 5).

2.2. Design Process Data Model:

Based on our multi-layered design process model, we propose the 'design process data model' (Figure 3). The model has been developed to represent the product design process and to structure the data so that it is prepared for analysis. The model's components are described below:

Product Data Model: captures the product's data during the design process. The product model results from this process.

- **Product Model:** the aggregation of the objects which evolve over time. The product model thus contains the list of states that identify the state of the product model in specific snapshots along the process.
- **Product Model State:** is transformed by the operations into another product model state. Different states of the product define the different versions of the product model.
- **Objects:** are created or modified by process element execution. Object attributes consist of the *ID* of an object which makes it a unique entity and helps to trace its changes (line, point, sketch, etc.).
- **Object State (O(j)i):** The object state is transformed by the operations into another object state. Each state refers to an object, and an object

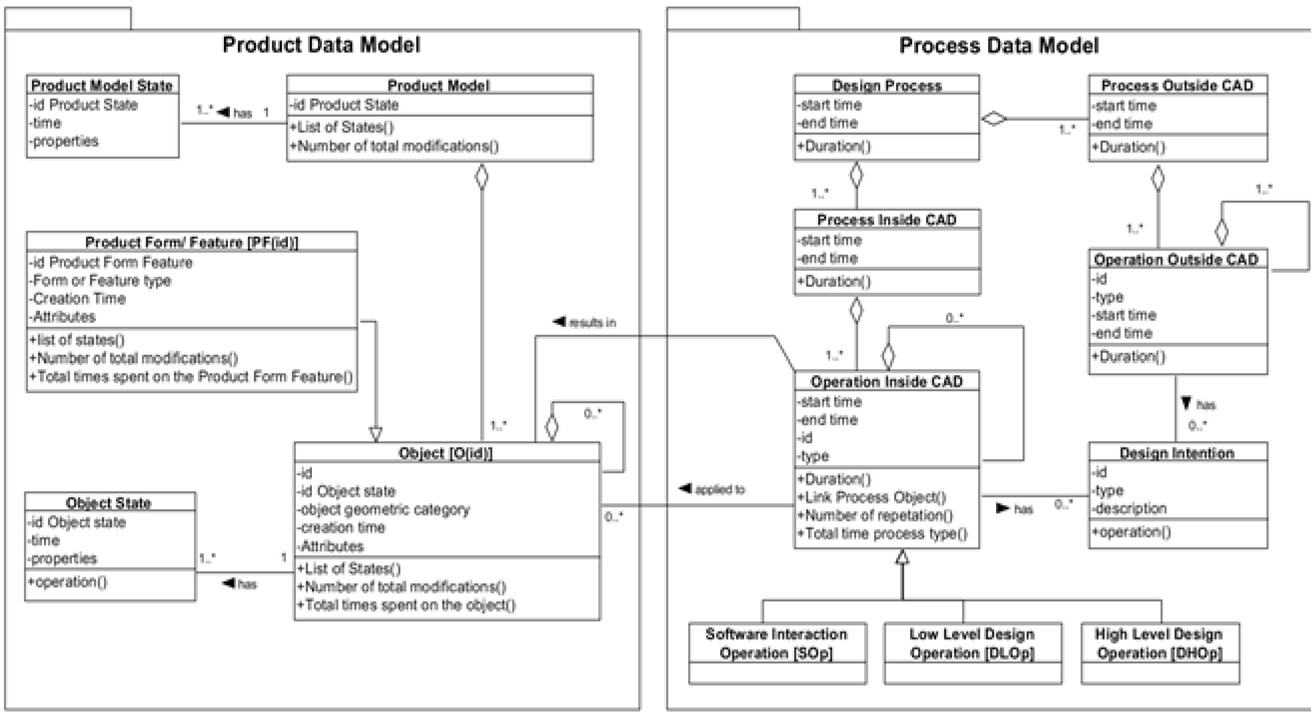

Figure 3 Design process data model

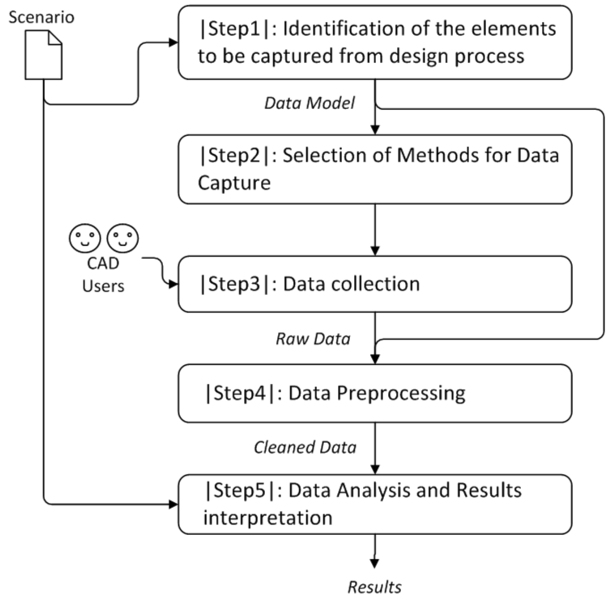

Figure 4 A structured approach to capture how designers use CAD software

has several states. The list of the pairs of objects' states and operations maintains a detailed history that includes information about how and when the object state has changed, the type of operation on the object (creation, modification), details about the parameters of the operation (the new values of the attributes), the sequence of changes in the object that create its final state, and tracing the evolution of a design. The object state keeps a record of the evolving values for each of its attributes.

- **Product Forms/features (PF):** Product Forms/Features are high-level design objects (e.g. holes, pockets). It is more important to capture these than to capture the other types of objects.
- **Process Form/feature State ($PF(j)_i$):** Just as with the Object state, each state refers to a product form/feature, and a product form/feature has several states.

Process Data Model: captures the data of the process itself. The *'ID'* is the number that identifies the operation's or process' identity. '*Type'* characterizes the type of the operation. '*Start time'* is the time the operation or process is triggered and '*End time'* is the time that that operation or process is finished.

- **Design Process:** the aggregation of the Process outside CAD and the Process inside CAD. It has a start time and an end time, hence we can calculate the process duration. The result of the whole process will be the final state of the product model. Several additional types of information are stored on the process side.
- **Process outside CAD:** the aggregation of the operations outside the CAD.
- **Operation outside CAD (O_{Pi}):** all types of operations that happen during the design process outside the CAD and that have a direct or indirect impact on the design process inside the CAD (ex., reading the design brief during the design process).
- **Process inside CAD:** the aggregation of the different types of operations performed on the objects inside the CAD.
- **Operation inside CAD:** The operation inside the CAD creates a new object or modifies an existing object or simply manipulates the object for review. When it is applied to the object a new state of the object will be created. An operation can aggregate several operations.
- **Software Interaction Operation (SO_{Pi}):** This type of operation is related to user interaction at the software interface level. For example, a user can use a keyboard shortcut to do an operation, or access the menu to do the same operation. This type of operation is tool-dependent.
- **Low Level Design Operation (D_LO_{Pi}):** Low level design operations are performed according to the sequence of the software operations. This type of operation is tool-independent and is characteristic of the key design operations. It supports direct creation (Line, Rectangle, Ellipse, Arc), modification (Undo, Again, Cut, Copy, Paste, Select All, Duplicate, Group, Push Back, Bring Front), set up (Font, Size, Style, Grid, Ruler, Scale, Snap, Browse) and manipulation (Rotate, Flip, Zoom) of objects.
- **High Level Design Operation (DHOPi):** The high level operations aggregate low level design operations into higher level units that focus on the design objects at the level of forms and features.
- **Design Intent:** Design intent is the thinking behind the user's operations and shows why users have done a specific operation.

When the actor wants to use the proposed model for a specific purpose, based on his/her requirements, s/he should create a sub model of the proposed model. The model has the potential to provide multiple levels of abstraction. The customization of this model based on the scenario requirements will help to capture the data elements of the scenario for further analysis.

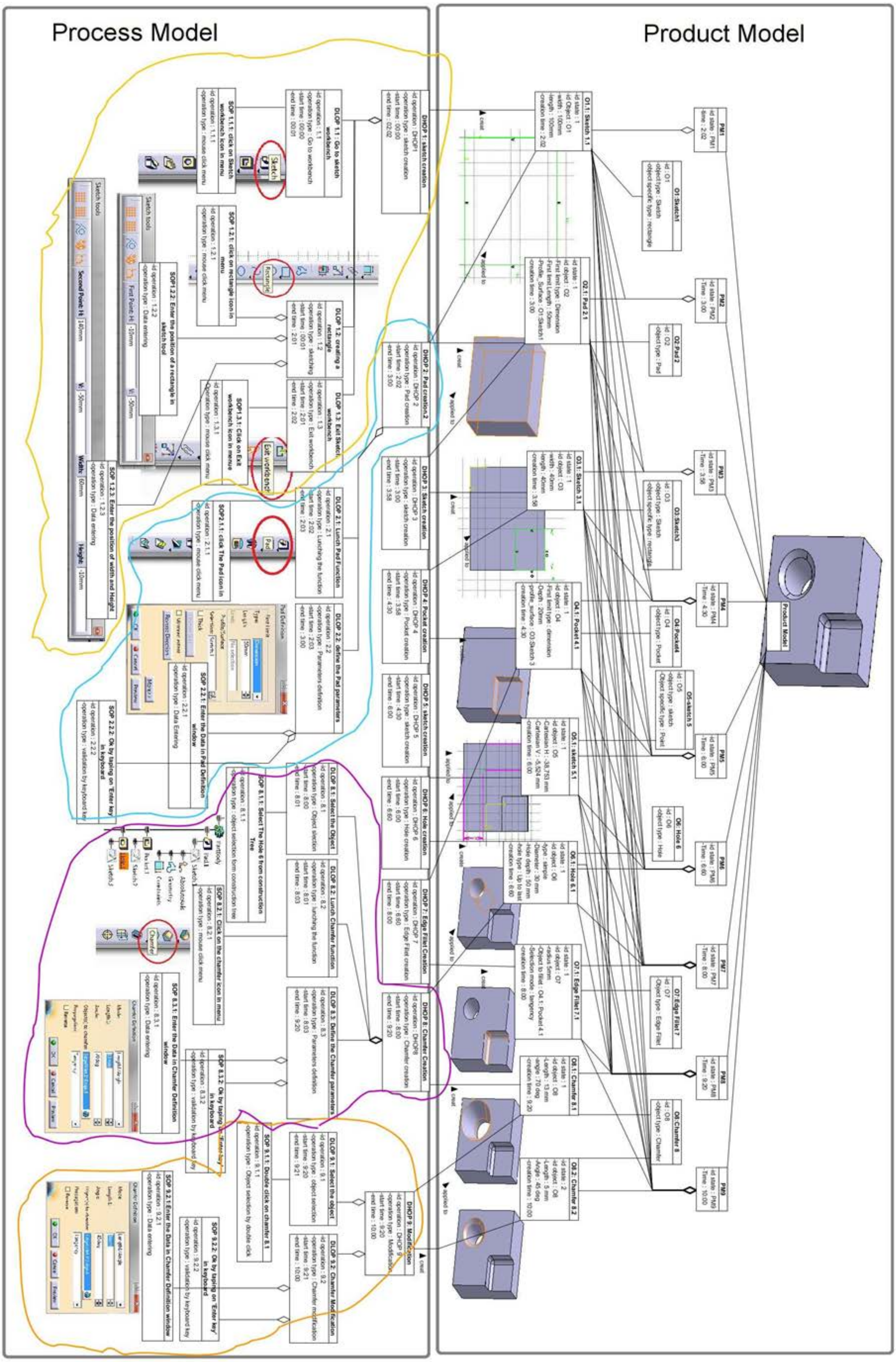

Figure 5 A model of figures 2 and 3 adapted to capture the data of the part in figure 1

Figure 5 illustrates the data model for the part introduced in figure 1. This figure illustrates the data model in a layered base format, which will help to better understand the multi-layered design process model and the design process data model.

3. PROPOSED APPROACH

We propose an approach (Figure 4) which leads the actor to reach his/her objective associated with his/her scenario. This approach consists of five main steps, developed in the following subsections:

3.1. |Step1|: Identification of the elements to be captured from the design process

First of all, the actor should identify which elements of the process should be captured, based on the scenario objectives. For example, is the time of the operations required or only the sequence of the operations?

The models proposed in the previous section will help the actor in this identification.

3.2. |Step2|: Selection of Methods for Data Capture

In this step the actor should select the appropriate method for data collection based on the result of the previous step. We have identified three main methods for data collection:

1- **Screen Video Capture:** screen recording software records the screen activities in real-time. This creates videos of whatever the user is doing on his/her desktop. It is intuitive and easy to use. The video data preserves the temporal and sequential structure of the user activity inside the CAD software. By slowing down and speeding up a video recording an actor can see all of the events. Despite the power of video to capture events, it is time-intensive to collect, review and analyze. As a consequence, the analysis tends to focus on short segments of video data at a micro analytical level [15].

2- **Macro Recording (CATIA):** The macro file or the log file corresponds to the procedural model. Macro files record the sequence modelling commands. In other words, a macro file is the history of the modeling commands issued by a designer. Most commercial CAD systems support macro file formats, but macro file formats are different for each CAD system. For example, the Macro file of CATIA is written as Visual Basic codes. CATIA records the names of the entities used during the modeling process and uses the 3D coordinates. To record entity names, CATIA uses a topological naming method [16]. However, the current version of Macro CATIA does not capture the time and software interaction operation.

3- **Think aloud protocol:** Much of the design process is a mental process; the sketches and drawings that form the visible record of designs do not disclose the underlying processes by which they were created. By visualizing the previous methods an actor can capture the actions of a designer performing a design task. In order to capture the mental process, the designer is encouraged to think aloud. However, this process may distract other designers during the design process.

Table 1 represents a summary comparing the three data collection methods. Each method has its advantages and disadvantages. During this step, the challenges are the tradeoffs between them, the quantity and quality of the information represented and the time needed to capture process and represent this information.

Table 1 Comparison between the three Data Collection methods

Process Data Model		Screen Video Capture	Macro Recording (CATIA)	Think aloud protocol
Product	Product Model	Yes	Yes	-
	Product Model State	Yes	Yes	-
	Object	Yes	Yes	-
	Object State	Yes	Yes	-
	Product Form or Feature	Yes	Yes	-
Process	Operation Outside CAD	Yes	-	-
	Software Interaction Operation	Yes	-	-
	Low level design Operation	Yes	Yes	-
	High level design Operation	Yes	Yes	-
User intent		-	-	Yes
Time		Yes	-	-
Bother the user during design process		-	-	Yes
Automatic way of recording the data		Yes	Yes	-

3.3. |Step3|: Data Collection

In this step, using the selected method for data capture, the data is collected from the target CAD users following the identified scenario.

3.4. |Step4|: Data Preprocessing

The raw data collected in the previous step should be cleaned to be prepared for data analysis. Depending on the data format, different processes and tools are required for this data cleaning. For example, the log-file of the CAD software saves the information related to the design process, but the text format nature of the log-file makes it difficult to use, as it requires a cleaning algorithm to be run on the macro file to prepare it for data analysis. Video files will also need to be reviewed and analyzed to extract the data in an appropriate format ready for analysis.

3.5. |Step5|: Data Analysis and Results Interpretation

Depending on the scenario, the data is analyzed in different ways. The actor should identify how this data can become meaningful to fulfill the requirements of a scenario. For example, if the objective is to compare the design process of two users, then, depending on the data type, the actor may create a graph of the user process and then compare it with graph of other users with comparison algorithms. Alternatively, he/she may just want to compare the two users based on statistical data.

4. CASE STUDY

In this case study, the objective is to compare the design processes of users that have created the same geometry. This will help to identify the users' level of expertise in using the CAD software as well as extract the hidden patterns of the processes.

4.1. Case Study |Step1|: Identification of the elements to be captured from the design process

We choose to focus on the operation inside the CAD and so we are not interested in the design objective or in the reasons why a user performs a specific operation. Figure 6 demonstrates the highlighted parts of the general model (Figure 3) that we wish to capture.

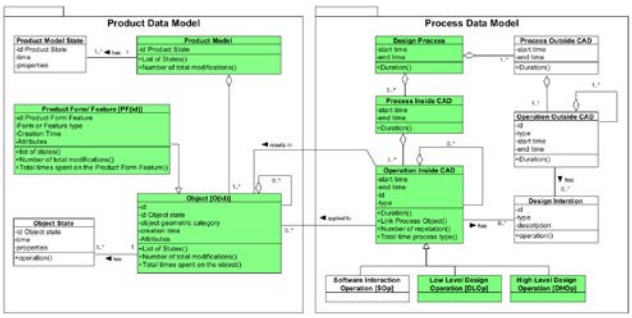

Figure 6 The model of figure 3 adapted for capturing data in our case study

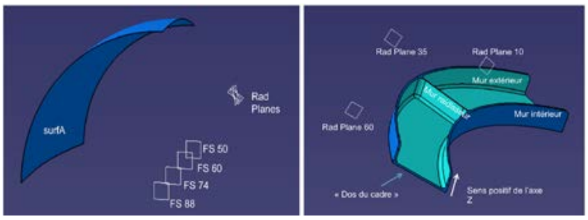

Figure 7 left: the initial model, right: the completed model

4.2. Case Study |Step2|: Selection of Methods for Data Capture

We selected the 'Screen Video Capture' method to capture information about user interactions with the CAD software. The user's screen has been captured while the user was designing the part. The reason for selecting this method is that we are specifically interested in comparing several users based on the time factor; i.e. the time a user spends on each operation, the correlation of the process time with the use of specific operations, etc. As mentioned above, the current version of Macro CATIA does not allow the time value to be captured; we can only see the notion of time in terms of the operations' sequences.

4.3. Case Study |Step3|: Data Collection

In the proposed scenario, we asked four CATIA users to design the same part based on the design brief given to them. The selected part was a piece of an aircraft structure (figure 7- right). This piece is derived from a subject that was given to engineering students. The user starts the modeling from a base part containing the plane and the surfaces (figure 7- left). The basic elements (such as the plane for the construction of references and the reference surface) have been made in advance to reduce the users' workload.

Each user's computer screen was captured during the modeling. Each user was in front of a single screen computer, in which CATIA V5 had been installed. The users were free to use methods and modeling tools of their choosing. The motivation for limiting the users to a single screen is that it allows the time that the user spends reviewing the design brief to be recorded. Hence, we can isolate this time from the total modeling time in order to calculate the effective modeling time.

4.4. Case Study |Step4|: Data Preprocessing

The video was viewed with care and the identified data elements and corresponding attributes (see

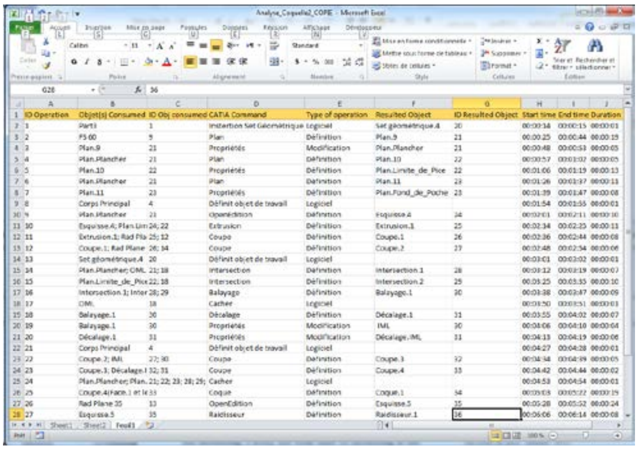

Figure 8 The raw data captured from the video and entered in an Excel file

Figure 6) were captured and entered in an Excel file (Figure 8).

4.5. Case Study |Step5|: Data Analysis and Results Interpretation

The collected data is analyzed in this step to make a comparison between the four users.

We defined several criteria for the comparison. The limitation that we encountered in comparing users was the "matching" of objects. We cannot say whether object cut.12 of user A corresponds to object cut.12 of user B, since they probably realized their cuts in a different order. We can however compare the number of created objects or the time spent by object type.

The summary of the results is presented in Table 2. These four users have achieved the same part; it can be concluded based on the comparison criteria that user 4 has the most effective modeling process. Firstly, his total modeling time is the shortest. He also made the least modeling operations (only 33) and created fewer objects to achieve the final model (only 20). He also demonstrated that he knows the subject very well, since he only spent 3" on the brief.

Table 2 Comparing the design metrics of four users

Criteria	User1	User2	User3	User4
Total time of the design Process	30:60	41:58	17:40	07:46
Total time of the design Process outside the CAD software	01:27	02:44	02:01	00:03
Total time of the design Process inside the CAD software= Total time of the design Process- Total time of the design Process outside the CAD software	29:33	39:14	15:39	07:43
Total time of the effective design	11:02	12:45	08:40	04:44
Ratio= Total time of effective design/ Total time of the design Process inside the CAD	37,43%	32,50%	55,38%	61,43%
The average time for an effective high level design operation	00:08,2	00:07,6	00:12,1	00:08,6
Total number of high level design operation	81	101	43	33
Total Number of Created Objects	35	55	21	20

The average time for an effective high level design operation allows us to measure the average time spent by a user in performing an operation. It is interesting to observe that for user 4, this time is not the shortest. User 2 is the one with the shortest average time, for an effective high level design operation. This would differentiate between two types of users: fast users who made many operations quickly, but whose operations are not always good, in the sense that they need to go back to delete or change objects; and slower users that spend more time for each operation and at the end will have made fewer operation to achieve the same geometric model.

Overall, the total modeling time increases with the number of actions. The number of objects increases too, which is quite logical because in general, each action creates an object.

Similarly, we note that user 2, the one with the longest modeling time and the worst efficiency ratio, has spent more time than others in the Sketch creation workshop, perhaps this is related to the user's skill level in using CATIA. Poor knowledge of the CATIA software would result in spending more time to realize effective actions. We can assume that user 2 is familiar with the CATIA software but that he/she may be less familiar with this type of part and so does not yet have a good working methodology for this type of part.

We defined another criterion: the ratio between the effective modeling time and the effective modeling time + idle time (= total modeling time - time outside the CAD). This ratio in some sense shows the efficiency of the user in using the software. Based on the time a user spends in front of the CATIA interface, this ratio provides information on the percentage of time during which he/she has taken actions which had a direct impact on the 3D model. It is still user 4 who has the best performance on this criterion, with over 60% efficiency in his/her use of the CATIA software. To achieve a ratio 100%, a user would have to make all of their actions one after another, without ever moving the model in space and knowing all the keyboard shortcuts for each CATIA command. It is therefore impossible to reach 100% for a human user. However, playing the macro part

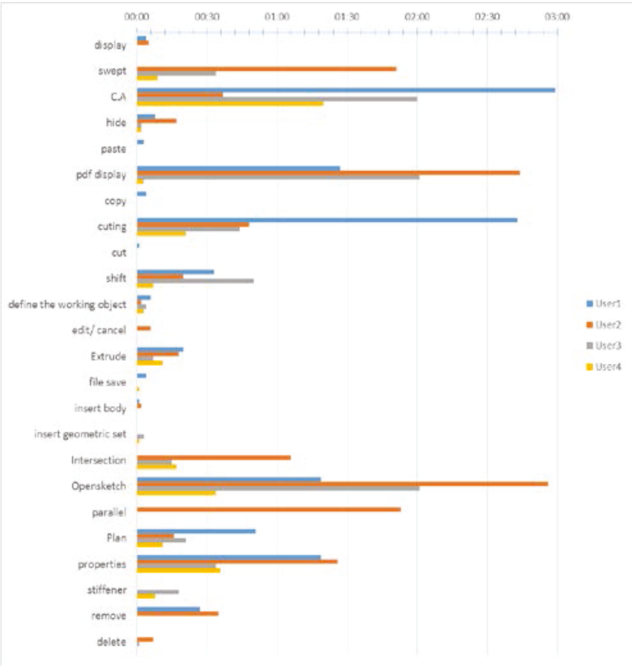

Figure 9 The total time spent on each type of operation in CATIA for four users

of a pre-recorded process, a 100% ratio could be reached.

Figure 9 illustrates the total time spent on each type of operation in CATIA for the four users. It allows us to find explanations for the differences observed between users. For example, users 3 and 4 are the only two users who used the "stiffening" function of CATIA software, they are also the two users who had top performers (55% and 61% efficiency by the ratio defined above), and they also had smaller total modeling times and workforce. Thus we can assume that the use of the stiffening function was particularly suitable for the modeling of this part.

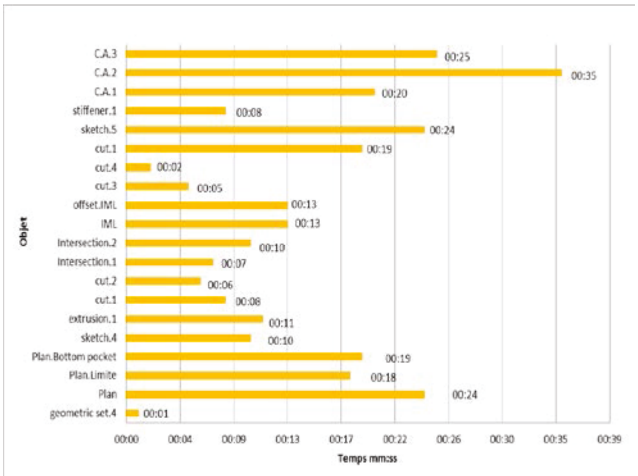

Figure 10 Total time spent on each object for user 4

The comparison gives us another image of the modeling process. We can view criteria other than those usually investigated. By analyzing the captured information, we can identify the trends and modeling habits of some users or of a group of people. (Example: User 1 uses the "copy - paste" and renaming function more often than other users).

Figure 10 illustrates the total time user 4 spent on each object. Such analysis can help to identify the difficulties that users have with specific types of objects. This time analysis can also be used to facilitate the creation of or the modification process of such types of objects, as comparing the time spent on specific types of objects between different users and comparing their processes helps to discover the most effective process for specific types of objects.

5. DISCUSSION AND CONCLUSION

In design companies, capturing process information during a product's design would help to better understand the design process and the sources of difficulty and mistakes. Making such information available for designers could then help to increase their productivity. In design education, collecting information on how design students generate geometric representations would allow teachers to identify areas of misunderstanding, and improve the education process by representing the optimal way of working.

A construction tree only represents the final outcome of the modeling process and not the path taken by the user to get to that final result. Hence, it has limitations in terms of displaying the users' working sequence and design processes. In this paper we propose a framework (models plus an approach) to support the analysis of how designers use CAD software to generate geometric representations. The proposed models help to track designer's operations, generating a stream of tool events while keeping track of the changes in the state of the product data.

Through a case study, we illustrated the different steps of the proposed approach. The CAD software used for this study is the CATIA software (V5R21) developed by Dassault Systèmes, however the results of this study can easily be generalized to other CAD software. We selected the 'Screen Video Capture' method to acquire information about user interaction with the CAD software, as video recordings make it possible to capture temporal sequential interaction. Video screen capture can lead to the collection of large amounts of rich data. However, video takes time to watch and review and can be difficult to meaningfully summarize. One minute of video takes an average of 8 minutes to fully extract the data (this time includes the time it takes to fill out the timetable associated with a video). Moreover, this data extraction is manual; it is not immune to a mistake when entering data in Excel or in the omission of an action. Hence, using video for large-scale campaigns (for hundreds of users) is not reasonable. Scaling up will require investigating automatic methods for capturing the design process. The main limitation of the current version of the macro file is the lack of time values for the operations. This study allowed us to have an idea of the types of conclusions that could be drawn from a comparison among four CAD users.

Capturing the design process and using artificial intelligence methods can provide automated guidance for those users unfamiliar with the software. A completely captured design process may also lead to a modeling process fully realized by the software; the user would only have to insert the characteristics of the geometric elements he/she wishes to build. This will require a detailed study of the geometric characteristics of the model, the identification of certain common geometries and their appropriate classification.

REFERENCES

[1] Myers, K. L., Zumel, N. B., & Garcia, P. (2000). "Acquiring design rationale automatically". AI EDAM, 14(02), 115-135.

[2] Chester, I. (2008). “3D-CAD: modern technology–outdated pedagogy?”. Design and Technology Education: an International Journal, 12(1), 7-9.

[3] Bhavnani, S. K., Garrett, J. H., & Shaw, D. S. (1993, June). “Leading indicators of CAD experience”. In CAAD Futures, Vol. 93, pp. 313-334.

[4] Nomaguchi, Y., & Fujita, K. (2013).“Knowledge representation framework for interactive capture and management of reflection process in product concepts development.”*Advanced Engineering Informatics*, *27*(4), 537-554.

[5] Ishino, Y., & Jin, Y. (2002). “Acquiring engineering knowledge from design processes”. AI EDAM, 16(02), 73-91.

[6] Ishino, Y., & Jin, Y. (2006). "An information value based approach to design procedure capture". Advanced Engineering Informatics, 20(1), 89-107.

[7] Sung, R., Ritchie, J. M., Rea, H. J., & Corney, J. (2011). "Automated design knowledge capture and representation in single-user CAD environments". Journal of Engineering Design, 22(7), 487-503.

[8] Sivanathan, A., Lim, T., Ritchie, J., Sung, R., Kosmadoudi, Z., & Liu, Y. (2015). The application of ubiquitous multimodal synchronous data capture in CAD. Computer-Aided Design, 59, 176-191.

[9] Lee, J., & Lai, K. Y. (1991). “What's in design rationale?”. Human–Computer Interaction, 6(3-4), 251-280.

[10] Potts, C., & Bruns, G. (1988, April). "Recording the reasons for design decisions". In Proceedings of the 10th international conference on Software engineering (pp. 418-427). IEEE Computer Society Press.

[11] Liang, Y., Liu, Y., Kwong, C. K., & Lee, W. B. (2009).“A design rationale representation model using patent documents”. In Proceedings of the 2nd international workshop on Patent information retrieval (pp. 57-64). ACM.

[12] Zhang, Y., Luo, X., Li, J., & Buis, J. J. (2013). “A semantic representation model for design rationale of products”. Advanced Engineering Informatics, 27(1), 13-26.

[13] Tang, A., Jin, Y., & Han, J. (2007). “A rationale-based architecture model for design traceability and reasoning”. Journal of Systems and Software, 80(6), 918-934.

[14] Myers, K. L., Zumel, N. B., & Garcia, P. (1999). Automated capture of rationale for the detailed design process. In Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference (pp. 876-883). American Association for Artificial Intelligence.

[15] Jewitt, C. (2012). "An introduction to using video for research". NCRM Working Paper. http://eprints.ncrm.ac.uk/2256/

[16] Choi, Guk-Heon, Duhwan Mun, and Soonhung Han. "Exchange of CAD Part Models Based on the Macro-Parametric Approach." International Journal of CAD/CAM 2.1 (2002): 13-21.

[17] Bracewell, R., Wallace, K., Moss, M., & Knott, D. (2009). Capturing design rationale. Computer-Aided Design, 41(3), 173-186.