Capitalizing and structuring design knowledge in an SME environment

Adel Amin AMMAR, Dominique SCARAVETTI, Jean-Pierre NADEAU

I2M, UMR CNRS 5295, IMC department
Arts et Métiers ParisTech
Esplanade des Arts et Métiers
33400 Talence, FRANCE
Tel. +33 5 56 84 54 22
dominique.scaravetti@ensam.eu

ABSTRACT. Small companies can find it difficult to preserve their knowledge, and also to structure a design process. A design methodology is proposed, based on design knowledge reuse and suitable for developing new manufacturing processes in an SME context. This paper describes a knowledge structuring and capitalization method, where a functional description is applied. The purpose is to capitalize technical solutions and the components used to carry out a given function, and to build a knowledge base that could be reused when designing new manufacturing processes. In this way, the time spent on research into design concepts can be reduced.

Components are identified using the Converter-Transmitter-Operator-Control classification, based on describing the functional flow path in terms of energy. Produced and induced effects associated with the components are highlighted, by identifying the relevant conjugate variables for the functional flows. The choice of solutions in the reuse phase is thus facilitated by considering these effects. In addition, a task decomposition tool has been developed to simplify the describing of existing manufacturing processes.

Existing knowledge capitalization methods proved unsuitable for an SME context. Based on the proposed approach, we applied our capitalization method in an industrial context, with the processes used by our partner company, which had never previously capitalized its design knowledge.

KEYWORDS: design knowledge capitalization, functional base, design methodology, SME, task analysis.

1. Introduction

Knowledge represents a very volatile intellectual capital for a company. Small and medium-sized enterprises are considerably weakened whenever one of their staff members leaves, as this results in a loss of knowledge [Wong et al., 2004] [McAdam et al., 2001]. For these small businesses, the loss could be lessened if they have been able to capitalize this knowledge and the skills of the person concerned. By storing earlier experiences, the company gains value and avoids repeating any mistakes that have already been made. The purpose of capitalizing the knowledge contained within projects is to ensure that the experience of all the participating individuals and groups can be made more widely available [Michellone et al., 2000]. It is a resource for those who have collaborated and who will then be able to recall the significance of certain situations and also for others who can find ideas for exploring new pathways and justifications of their design decisions [Kuffner et al., 1991].

Capitalizing on produced knowledge and reusing it in other projects is becoming an essential challenge for businesses. Knowledge, even more than capital and physical resources, has become the essential ingredient for creating value [Rikowski, 2003]. According to Barney, a company's resources are its assets, capabilities, organizational processes, information and knowledge that it controls and which enable it to conceive and implement its strategies [Barney, 1991].

Knowledge and design are very closely linked: The recollection and application of knowledge can be considered as a straightforward and practical design process [McMahon et al., 1998]. The sharing of design knowledge can be expected to improve the design process drastically [Kitamura et al., 2004]. However, in the course of the design process, a very large amount of information and knowledge is exchanged between the actors involved in the project. In order to save time searching for data and information, this knowledge should be accessible to members of the design team and so needs to be capitalized in a format that can be updated as and when required for different projects. This accessibility is critical for small firms as they rarely have their own design department. Different methods of knowledge management exist but they can be very cumbersome to put into practice in a small firm with few resources and qualified staff. These firms are therefore faced with the problem of how best to capitalize and store their knowledge. They need an easy and intuitive way of storing and reusing their design knowledge.
Matta et al. have defined the different phases of knowledge management, which consist of extracting knowledge, formalizing it, capitalizing it, then making it accessible [Matta et al., 2001]. The work presented in this article was carried out in collaboration with a small company of about thirty employees which manufactures footwear. It is not necessarily typical of all SMEs. The company wanted to set up a project which would highlight innovation and knowledge management. Their aim was to put in place a whole design methodology for new production processes, based on the reuse of existing knowledge.

In this article we focus on presenting the procedures we recommended for extracting, formalizing and structuring knowledge associated with the skills and know-how that was available in the company. Our aim was to produce a methodology to be used as a guide for building a knowledge base. In the first part of the article we analyze several knowledge capitalization methods that can be found in the existing literature, looking in particular at those which deal with design projects. The basic element in our capitalization method is the elementary function. An elementary function is one that cannot be decomposed further. For each function, we capitalize all information relevant to carrying it out. The procedure for extracting and structuring knowledge within an SME is described in the second part of the article. The elementary function is commonly and easily used within the SME and by using

computer software, information can be retrieved for reuse quickly and easily, in a way that is adapted to the context of a small firm. When the required function is entered, this system will propose various solutions for this function, something that the designer does not usually have time to do. In the last part an industrial application of knowledge capitalization is described using an example from our partner SME.

2. Knowledge capitalization

2.1. Knowledge typologies

During the knowledge capitalization phase, a distinction must first be made between tacit and explicit knowledge [Polanyi, 1966]. Tacit knowledge is difficult to formalize and communicate as it has a personal side to it; this knowledge is very closely linked with practical work, like an acquired knack for doing something. An important aspect of knowledge management is to formalize this tacit knowledge, in order to make use of it and ensure that it can become operational throughout the organization [Nonaka et al., 1995]. Explicit knowledge, on the other hand, is easy to transmit as it can be formalized through a formal language. This knowledge is slightly removed from the practical side but it is nevertheless tangible (data, plans, procedures, models, etc.).

A second distinction is made between knowledge and know-how. For Ermine, static knowledge models the objects and designs of the domain being considered, whereas dynamic knowledge models the strategies for using this static knowledge to solve a problem [Ermine et al., 1996]. Thus static knowledge represents “knowledge” and the dynamic component of this knowledge represents “know-how”.

Tan and Libby distinguish between technical knowledge and managerial knowledge [Tan et al., 1997]. Technical knowledge is knowledge that is useful to a person or a group of people for carrying out a task. Managerial knowledge, however, concerns the company’s organizational and strategic activities. Uluoglu defines procedural knowledge, which is used to set out the rules or conditions for carrying out a task [Uluoğlu, 2000]. This vision is shared by Aidi who defined knowledge as being set rules, cases or constraints [Aidi et al., 2008].

In the case of our partner SME, the types of knowledge to be capitalized are mainly tacit knowledge about the way processes are carried out, product design, and solution concepts used in footwear manufacturing processes. Many of the production systems have been designed and produced outside the SME and so the company does not necessarily have design data. Some time must therefore be spent on observing existing procedures, to formalize both concretely and technically the way in which the functions are carried out.

2.2. Capitalization methods

Several capitalization methods exist, inherited from knowledge engineering [Holsapple et al., 2001] [Lai et al., 2000] and which propose models and methodologies (interviews, document analysis, etc.). Baxter et al. propose a review of design reuse methods [Baxter et al., 2007]. According to [Aidi et al., 2008], design processes imply capitalization, sharing and reuse of knowledge about numerical simulation. Many knowledge reuse systems are not usable in early design stages, because they are focused on geometrical data and dedicated to detail design phase. Such systems are not appropriate in our case, because the company needs to design new processes, not only to redesign existing manufacturing processes.

Rasovska has also identified knowledge management methods and tools in design projects [Rasovska et al., 2008]. From these, we studied four methods which seemed to us to deal specifically with knowledge capitalization associated with manufacturing processes: first, the aim of the MASK method (Method for Analyzing and Structuring Knowledge) [Matta et al.,

2001] is to model knowledge, to formalize it in graph form which clearly shows the interactions between each element of knowledge. Second, the basic principle of the SAGACE method [Penalva, 1993] consists of modeling the static knowledge that describes a production system. This modeling is based on three types of vision: functional, organic and operational. Third, using the CommonKADS method [Schreiber et al., 1999], the entire knowledge acquisition process can be dealt with, from knowledge collection to the development of a basic knowledge system. Lastly, we looked at the Componential Framework method [Steels, 1993a] [Steels, 1993b] which is based on modeling knowledge from three perspectives: task, information and method.

We will see what aspects of knowledge are common to these four approaches and also any features that are found in only one model. Each approach presents at least a description of the activity in the form of tasks. The MASK method proposes a downward analysis of the functional type where each activity is hierarchically decomposed. This decomposition into a tree structure refines tasks recursively from the top down towards the more detailed sub-tasks. The ordering of the tasks is also specified.

In Componential Frameworks, the tasks are also decomposed into graph form, using a tree structure called the "task structure". Models specify the knowledge that has been consulted and constructed to complete a task and show how this knowledge is used to carry out each task. With SAGACE, the task and actors are included in the operational vision, whereas the functions, design constraints and objectives are included in the functional vision. CommonKADS defines the overall organization of the tasks, input and output, pre-conditions and performance criteria. It also describes the human or computerized agents involved in carrying out the tasks, and the resources and skills required. Man-machine communications are specified. Finally, details are given about company organization, describing major functions.

Another feature of the MASK method is that it integrates knowledge evolution to provide a global assessment of the changes that have led up to the knowledge being in its present state. Also, using a retrospective analysis, the evolution of the main objects or concepts of the system can be visualized. A genealogical tree traces items that have appeared and disappeared, reasons for evolution, and also the positive and negative elements that each generation contributes.

Lastly, when describing solutions that are applied, MASK specifies the design structure put in place to carry out tasks or functions. SAGACE talks of an organic vision. The CommonKADS design model corresponds to the technical specifications of the system. In addition to this, MASK characterizes physical phenomena by describing interactions between two systems via a flow (of matter, energy or information).

2.3. Synthesis and needs

All these methods can be applied to design as they enable different types of knowledge to be manipulated. Using the MASK, Componential Framework and CommonKADS methods, several knowledge typologies can be defined: information, context, importance, task, method, domain model and ontology. These methods focus on representing a process. With SAGACE, functional, organic and operational aspects can be represented in terms of the result of a project. Thus all these tools describe systems, even complex organizations. They do this by decomposing a process into tasks and defining the sequence that links the tasks together. However, the time required to carry out the tasks, the description of the context surrounding the tasks (objects, users, etc.), the conditions required for passing from one task to another, are not always given.

In our case, the knowledge to be capitalized was linked with manufacturing processes. In order to describe and capitalize an entire process, it had to be possible to retrace the ordering

of tasks and elementary functions over time, to determine sequencing and transitions between them, and also to specify whether a task was allocated to human operators or to the machine. Man-machine interaction phases had to be shown. Finally, the company wanted to capitalize the technical solution employed, the components or sub-units used, and the energy required to carry out each elementary function,.

In the past, it was not possible to exploit previous designs efficiently as there was no methodology to structure them and no structured database [Shahin et al., 1999]. We therefore developed a knowledge capitalization and structuring procedure, specifically tailored to the company's needs.

In this article, we show that the procedure we proposed, based on task analysis of existing procedures, enabled us to draw up a list of the actions carried out by the operator, to extract the physical effects present and associate them to design criteria and variables. This capitalization and analysis was structured around functional logic and resulted in the construction of a knowledge base which had to be able to fulfill at least two objectives: storing knowledge associated with the SME's acquired information and know-how, and ensuring that, when needed, this knowledge would be accessible to the designers, so that they could reuse it. Since the company is small in size, with only limited resources, important factors such as ease of access to the database, availability of technical solutions and relevance of related information all had to be taken into account.

First, we describe the method used to analyze and capitalize existing processes in the SME. We then explain the procedure for extracting and structuring the capitalized knowledge. A case study is described to illustrate the application of this method.

3. Sequential analysis of manufacturing processes

Our starting point was the analysis of existing processes in the SME. This was based on two main sources:

- Individual skills linked with the knowledge and know-how of the company employees who provided qualitative information on what the process being studied consisted of. This information was capitalized through discussion and interviews with the different operators in the company and also by observing them at their work stations.

- The acquisition of data about the system itself, which provided information that was more quantitative, in the form of measurements. This data could be capitalized quickly when a technical file existed for the process being studied. If not, then it could take a long time to fully understand and analyze the system.

We define a process as being a succession of basic tasks, also called significant moments [Doré at al., 2007a]. In fact, the reason for modeling the process in this way is to show and describe the tasks to be completed in order to carry out a job. Thus, the detailed analysis of the tasks enables us to:

- Define the goal to be achieved by a task,

- Describe the mode (manual or automatic), the manner and the moment of execution of a task,

- Take any constraints into account (pre-conditions, response time, safety, etc.),

- Extract all the means, components, knowledge and information that may be necessary or useful to carry out a task.

For Baxter et al. [Baxter et al., 2007], knowledge can be stored in computer-based systems, in a variety of forms (text, images, rules, diagrams, etc.), but the knowledge format

must enable easy retrieval and appropriate application later. To ensure the capture and effective reuse of design information, it is essential to represent the knowledge appropriately [Ki Moon et al., 2009]. We propose a graph model in tree structure form to represent hierarchical decomposition during the conceptual design phase. This shows relations between functional modules and process modules. Both linguistic and parametric information are captured.

Processes are decomposed into tasks using the CTTE tool (Concur Task Trees Environment) [Mori et al., 2002]. The resulting graph shows the distribution of tasks to the operators or the system, thus it describes interaction between user and system. It also includes temporal information, such as the order and scheduling of tasks or recursive actions which are difficult to describe using functional analysis tools (e.g. FAST). Figure 1 shows an example of task decomposition for a heel/sole assembly process.

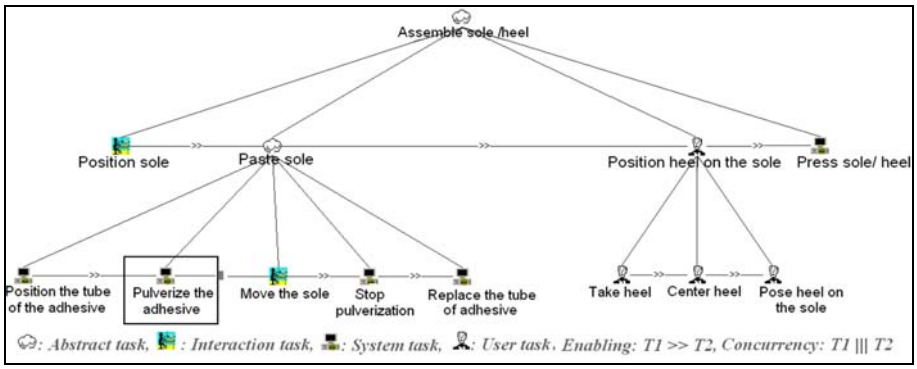

Figure 1. Functional decomposition of an assembly process

The decomposed tree structure of the tasks is read from the top down. We distinguish four types of task:

- User tasks, which are carried out by the operator,
- System tasks, which are carried out by the system,
- Interaction tasks which specify user interaction on the system via input means,
- Abstract tasks, of a high level of abstraction and which must in turn be decomposed.

Using this tool, we decomposed the tasks into sub-tasks or elementary functions. This decomposition gave a hierarchical representation and a capitalization of all the knowledge by moving from the most general (parent task) to the most detailed (sub-tasks or elementary functions).

In the task model, an elementary function is an elementary action that is not or cannot be decomposed. A function is carried out based on functional flows which concern the transit and transformation of energy, matter and signals (or information) [Bo et al., 1999] [Stone et al., 2000]. The move from one function to another is defined by transitions and control elements. These elements do not appear in the previous decomposition, but are represented in the Grafcet formalism: figure 2 shows the decomposition of the "paste sole" task from the process described in figure 1. Transitions determine the move from one function to another, with the control/command element allowing this transition to occur. This may be the result of material systems or it may be linked to the operator's somesthetic and sensorial receptors: A sensory space is a mix of sub-spaces such as hearing, sight, and touch [Doré et al., 2007b].

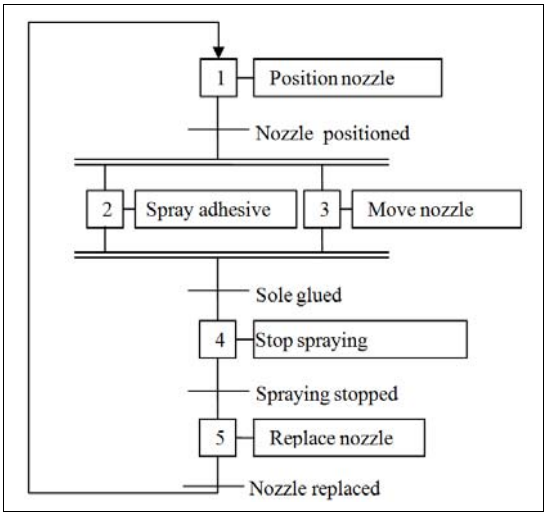

Figure 2. Grafcet chart of task of gluing (level 1) in the assembly process

We hope to improve representation of the functional and temporal decomposition proposed by CTTE by combining transition conditions.

For this reason, we now propose a new version of the task decomposition graph. The type of task is no longer defined by a logo as in figure 1 but by letters. Thus, the letters A, S, I and U represent respectively Abstract, System, Interaction and User tasks. At each decomposition level i, tasks are ordered according to their temporal orders j. Thus, we express as $A_{i,j}$ the abstract task of order j at decomposition level i. However, two specific situations that may arise when using this notation are as follows:

- Two tasks of the same type (e.g. abstract) at the same level, which happen in parallel, are expressed as follows: the first $A_{i,j}$ and the second $A'_{i,j}$.

- Two different tasks (e.g. an abstract task and a system task) at the same level which happen in parallel: the first is expressed $A_{i,j}$ and the second $S_{i,j}$.

In order to define transitions between tasks in the decomposition graph, we have defined a code to designate transitions, shown in figure 3. The transition between these two consecutive tasks is expressed as $i,X_{i,j},X_{i,j+1}$.

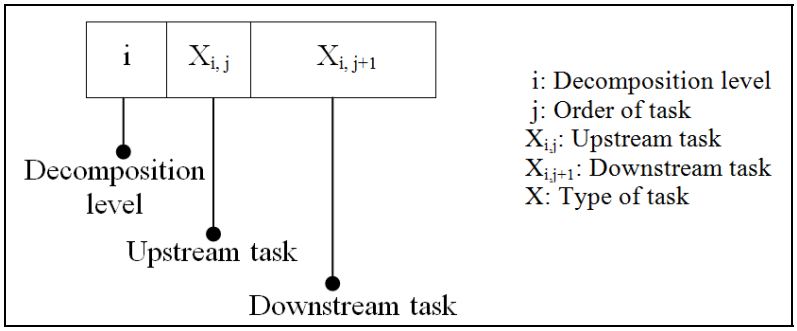

Figure 3. Designation code of transitions between two tasks

Based on this information, figure 4 shows our graphic representation of the functional decomposition of a process. Task allocation, temporal specifications, order of execution and transitions between tasks are all shown.

The transition to complete task Ai+1,k-1 is the transition upstream from the parent task Ai+1,k-1. This transition is written as: i,Si,j-1,Ai,j. The transition for controlling task Ui+1,k+1 and moving on to the next task is the downstream transition from the parent task Ui+1,k+1, in this case, the transition is expressed as i,Ai,j,Ui,j+1.

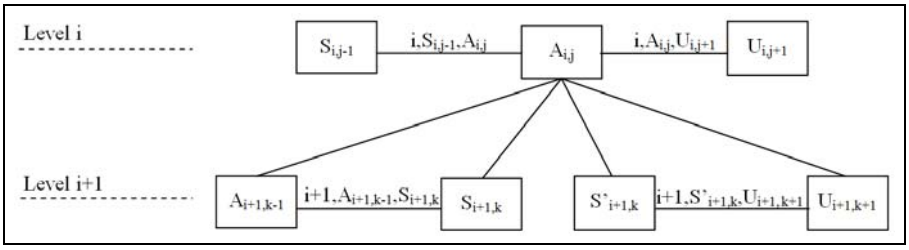

Figure 4. Decomposition of tasks taking into account task allocation, time, transitions

Thus, a production process can be archived, using this description which specifies the tasks required, the order in which they must be carried out, the person or machine to which they are allocated, and the transitions required.

By analyzing elementary functions, we are then able to capitalize all the relevant knowledge that makes up the knowledge base. To achieve this, we put in place a knowledge extraction and structuring procedure to analyze all the elementary functions of each task in a process.

4. Procedure for extracting and structuring knowledge

The purpose of a knowledge base is to model and store knowledge and thus ensure that it can be reused whenever the SME needs to design new processes. The extraction procedure should facilitate the emergence of the knowledge that has to be taken into account as the knowledge base is being formulated. In addition, raw data requires interpretation and restructuring to become reusable information.

In our structuring procedure, the aim is to find the link between functions and the components required to carry them out. This procedure is based on earlier studies [Sallaou, 2008], so that at each stage of the structuring procedure we are able to extract the relevant elements to put into the data base, to link them and prioritize them for future use.

4.1. Structuring functions

Based on the phase described above to analyze existing processes, we put in place technical functions and functions linked with operators' know-how. These functions were formulated using a verb and a complement [Hirtz et al., 2002]. Szykman raised the need for a standardized terminology in design artefact description, and proposed taxonomies of function and flow [Szykman et al., 1999]. A study has highlighted the need to consider both the verb and noun together [Ahmed, 2005], but also to combine a function taxonomy with other taxonomies to avoid loss of information.

We wanted to develop a standard vocabulary to facilitate the structuring process and ensure that when a term is used to describe a function it is unique and exhaustive. For this, verb and complement bases were used [Ammar et al., 2010]; however, these are not the main focus of this paper.

4.2. Description of components carrying out functions

Carrying out a function implies the transformation of energy. We use the first law of evolution of technical systems [Savransky, 2000] to define four essential elements. According to this law, carrying out a function depends on the transformation of energy (Converter), which is then transmitted (Transmitter), and an Operator carries out the action; the system is optimal when it includes a Control/command function carried out by a controller component. By using this classification, the identification of components that enable a given function to be carried out can be systematized.

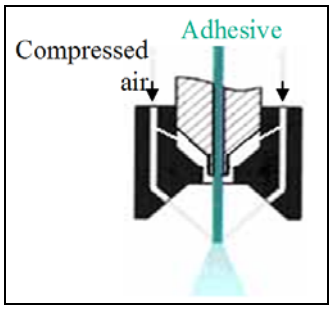

Figure 5. Operation of adhesive spray nozzle

For example, the function "pulverize the adhesive" (see box in figure 1) can be performed using a compressed air system at the nozzle: figure 5 shows the flow of adhesive at the centre of the nozzle, and the compressed air input which sprays the adhesive coming out of the nozzle. Figure 6 shows the Converter (compressor which transforms electrical energy into pneumatic energy), the Transmitter (flexible hoses) and the Operator (spray nozzle) used to carry out this function. Control/Command is achieved by controlling pressure on the Converter and the Operator.

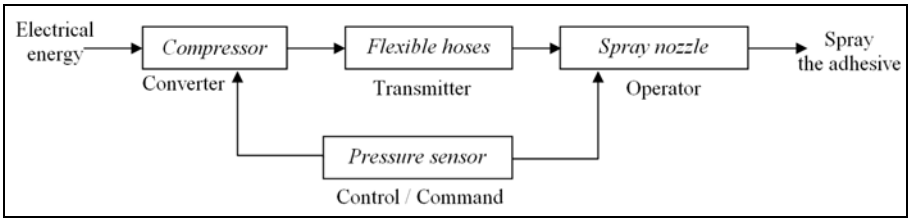

Figure 6. Analysis of the service function "Spray the adhesive"

4.3. Functional flows and relevant conjugated variables

A function is carried out by transforming energy. Thus, for each type of energy involved, we define the parameters that characterize both input and output energy; these we have called the relevant conjugate variables. Energy flow is produced between the state variable

and its temporal variable. Table 1 gives the corresponding relevant variables for some energy types, and energy flow is expressed.

After identifying these relevant conjugate variables we are able to specify the type of energy entering and leaving the component.

This definition of the relevant variables can also be used to deduce effects that are produced and induced in association with the components and with their interactions [Sallaou, 2008].

Type of energy	Relevant conjugate variables		Energy flow (power)
	Temporal Variables	State Variables	
Mechanical (Translation)	Speed (V)	Force (F)	F.V
Mechanical (Rotation)	Rotation speed (ω)	Couple (C)	C.ω
Hydraulic / Pneumatic	Volume flow rate (qv)	Pressure (p)	q_v .p
Thermal (Sensitive)	Capacity flow rate (q.Cp)	Temperature (T)	q.Cp.T
Thermal (Storage)	Flow rate (q)	Internal Calorific Value (PCI)	q.PCI
Electrical	Current (I)	Electrical potential (U)	I.U
Static Mechanical (Translation)	Virtual speed (V^*)	Force (F)	0
Static Mechanical (Rotation)	Virtual rotation speed (ω^*)	Torque (C)	0

Table 1. Examples of relevant conjugate variables

4.4. Produced effects / induced effects

In order to define and generate system architecture according to component assembly, we incorporate knowledge linked with produced and induced effects and also the physical models that result. A produced effect is one that is linked only to the conjugate variables defined earlier. Induced effects appear if there are elements that are in opposition to the produced effects due to system design. For example, if a component becomes distorted and if the adjacent component prevents it from moving freely, then thermo-mechanical stresses will appear (change in geometry).

Table 2 lists some produced effects with examples of induced effects, according to the conjugate variables involved. Produced effects are systemic and associated with types of flow. Induced effects will be selected according to the applications being considered. Knowing the induced effects and any harmful consequences in relation to adjacent components can enable the designer to eliminate a particular technical solution if he is presented with a choice.

State variables	Time variables	Produced effects	Induced effects
Effort (F)	Speed (V)	Strain	Play/Restraint/Stresses/Vibrations
		Friction	Wear/Heat transfer/Dilation/Retraction /Play /Restraint/Stresses /Warp
Pressure (P)	Volume flow rate (q_v)	Strain	Leaks/Stresses
		Friction	Dilation/Retraction/Play/Restraint/Stresses Pollution/Clogging
Temperature (T)	Capacity rate (q.Cp)	Heat flow	Dilation/Retraction/Play/Restraint /Stresses /Warp/Icing/Icing up
		Friction	Dilation/Retraction/Play/Restraint/Stresses Pollution/Clogging

Table 2. Examples of produced effects and induced effects [Sallaou, 2008]

For each component to fulfill its function correctly, it is imperative that its definition includes the calculation of its dimensions. In this phase of the process, first, any induced

effects can be overcome and, second, design solution performances can be assessed or a choice made between different solutions. For this reason, we capitalize not only the conjugate variables associated with component functions, but also numerical information (numerical values, intervals, etc.).

5. Case study: applying the capitalization procedure to the SME

The procedure presented above was applied to all processes used by our partner, which is a small footwear manufacturing firm.

Based on the elements listed earlier, a task analysis document was drawn up. Table 3 shows analysis of the "paste sole" task in the heel/sole assembly process. Line 2 corresponds to the elementary function "spray the adhesive" and describes a particular solution for carrying out this function. The components of this technical solution are identified from the Converter/Transmitter/Operator classification described in figure 6. Control is by means of a pressure sensor in the compressor and the spray nozzle. Using table 1, we can specify for each component the type of energy and also the conjugate variables that characterize input and output flows. Input energy in the compressor is electrical energy (U.I). Output energy is pneumatic energy (qv .p) which is transmitted by the hoses (transmitters) and then becomes the input energy for the spray nozzle. The flows of matter (compressed air and adhesive) which combine in the nozzle generate fluidic energy as they leave it. Information flows are also considered in our analysis documents to further simplify the job of choosing solutions in the re-use phase. In the case of this technical solution, information flows concern the pressure value at the entry to the spray nozzle, shown by the pressure sensor. Using the earlier identification of the energies involved, produced and induced effects can be determined from table 2. For both pneumatic and fluidic energy, the produced effects are the same: strain and friction. For the induced effects, based on the last two lines in table 2, we select those which are relevant to the solution being studied: in this case these are leaks in the components, dilation due to the heating of the air and the adhesive, clogging of the hoses and the nozzle and pollution due to the gluing process. When the designer has several technical solutions available for carrying out a function, then these elements will facilitate his decision-making process.

Innovation can come from new combinations of functions or of already known solutions. However, the base proposed is especially intended to help the designer reduce the time needed to search for technical solutions to fulfill a given function. When the base proposes solutions, the designer has to make a choice, while being helped by the functional requirement criteria.

Document No.: Date: Page:		**Analysis of task no. 2:** **Glue the sole**						
No.	**Function**	**Components**	**Control**	**Flow**			**Produced effects**	**Induced effects**
				Energy E/S	**Matter**	**Information**		
1	..							
2	Spray adhesive	C: Compressor	Pressure sensor	E:Electrical (U= 230V, I= 11 A) S:Pneumatic (qv= 433 l/min, p= 7 bar)	-Compressed air - Adhesive	Pressure entering spray nozzle = 2.5 bar.	-Strain -Friction	-Leaks, -Dilation -Clogging -Pollution
		T: Hoses		Pneumatic				
		O: spray nozzle	Pressure sensor	E: Pneumatic S: Fluid (qv = 0.1 at 28 ml/s, p= 0.1 at 3 bar max)				
3	..							

Table 3. Task analysis document for task 2 of the heel/sole assembly process

6. Conclusion

In this article we have presented a methodology for capitalizing and structuring knowledge relating to manufacturing processes, as applied to a small business. Our aim was to put in place a knowledge base that could be reused when designing new manufacturing processes in order to reduce the time spent on research into design concepts.

Existing knowledge capitalization methods proved to be unsuitable for our purposes: knowledge acquisition is not a dynamic process and does not take into account the time taken to carry out the different tasks of a process, nor the transition between tasks. Moreover, these methods are based on modeling knowledge across several different models, thus making them difficult to exploit in an SME where there are not sufficient qualified staff and where the manager would not have time to carry out this work himself.

For these reasons, we developed our own knowledge capitalization method which was able to meet the needs of our partner company. It is based on the hierarchical decomposition of processes into elementary tasks. This decomposition is represented in the form of a graph which takes into account how tasks are allocated to operators or to the system, interaction between user and system, time-related information such as the order and sequencing of tasks, and also control and transition elements between tasks. To facilitate knowledge structuring and ensure that descriptions are as exhaustive as possible, functions are formulated using a verb plus a complement and are then associated with components. The reason for this is to capitalize the technical solution applied and the components or sub-units used to carry out a given function. According to Ahmed, using function taxonomy could encourage novice designers to think in terms of the more experienced designers [Ahmed, 2005].

Components are identified using the Converter/Transmitter/Operator/Control CTOC classification, which describes the functional flow path in terms of energy. By using relevant conjugate variables according to the type of energy applied, all the produced and induced effects associated with the components can be structured. These are used at a later date to generate physical models.
On the basis of this procedure, we devised process analysis documents and then presented an industrial application of knowledge capitalization taking processes used in our partner SME as an example.

This procedure can be adapted to any company, although each company will of course have to enter the verbs and complements specific to their own business along with the associated technical solutions [Ammar et al., 2010].

This methodology was carried out with easy to use software, enabling the capitalized knowledge to be stored and shared. Knowledge formalization by designers is only required when updating the knowledge base. The software tool allows new functions or new solutions to be entered simply and easily. However, ensuring that this knowledge can then be appropriated and reused in the context of an SME, remains a real challenge.

7. Bibliography

Ahmed, S., "Encouraging reuse of design knowledge: a method to index knowledge", *Design Studies*, Vol. 26, No. 6, 2005, pp. 565-592.

Aidi, M., Gautier, R., Tollenaere, M., Pourroy, F., Maalej, A., "Steering numerical simulations by means of requirement engineering", International Journal Of Design and Innovation Research, Vol. 4, N°1, 2008, pp. 23-37.

Ammar, A.A, Scaravetti, D., Nadeau, J.P., "Knowledge reuse: towards a design tool", In *Proceedings of the 11th International Design Conference - DESIGN 2010*, Dubrovnik, Croatia, 2010.

Barney, J., "Firm resources and sustained competitive advantage", *Journal of Management*, Vol. 17, No. 1, 1991, pp. 99-120.

Baxter, D. , Gao, J. , Case, K. , Harding, J. , Young, B. , Cochrane, S. , Dani, S., "An engineering design knowledge reuse methodology using process modelling", *Research in Engineering Design* , Vol. 18, No. 1, 2007, pp. 37-48.

Bo, Y., Salustri, F.A., "Function Modelling Based on Interactions of Mass, Energy and Information", In *Proceeding of the Twelfth International Florida Artificial Intelligence Research Society Conference*, Orlando, Florida, 1999, pp. 384-388.

Doré, R., Pailhès, J., Fischer, X., Nadeau, J.P., "Identification of design variables and criterion variables towards the integration of user requirements into preliminary design", *International Journal of Product Development*, Vol. 4, No. 5, 2007a, pp. 508-529.

Doré, R., Pailhès, J., Fischer, X., Nadeau, J.P., "Identification of sensory variables towards the integration of user requirements into preliminary design", *International Journal of Industrial Ergonomics*, Vol. 37, No. 1, 2007b, pp. 1-11.

Ermine, J.L., Chaillot, M., Bigeon, P., Charreton, B., Malavieille, D., "MKSM, a method for knowledge management. Knowledge Management, Organization, Competence and Methodology", *Advances in Knowledge Management*, Vol. 1, 1996, pp. 288-302.

Hirtz, J., Stone, R., McAdams, D., Szykman, S. Wood K., "A Functional Basis for engineering Design: Reconciling and Evolving Previous Efforts", *Research in Engineering Design*, Vol. 13, No. 2, 2002, pp. 65-82.

Holsapple, C.W., Joshi, K.D., "Description and Analysis of Existing Knowledge Management Frameworks", In *Proceedings of 32nd Hawaii International Conference on System Sciences, IEEE*, Hawaii, USA, 2001.

Ki Moon, S., Simpson, T.W., Shu, J., Kumura, S.R.T., "Service representation for capturing and reusing design knowledge in product and service families using object-oriented concepts and an ontology", *Journal of Engineering Design*, Vol. 20, No. 4, 2009, pp. 413-431.

Kitamura, Y., Mizoguchi, R., "Ontology-based systematization of functional knowledge", *Journal of Engineering Design*, Vol.15, No. 4, 2004, pp. 327-351.

Kuffner, T.A., Ullman, D.G., "The information requests of mechanical design engineers", *Design Studies*, Vol. 12, No. 1, 1991, pp. 42-50.

Lai, H., Chu, T., "Knowledge Management: A Review of Theoretical Frameworks and Industrial Cases", In *Proceedings of the 33rd Hawaii International Conference on System Sciences, IEEE*, Hawaii, USA, 2000.

Matta, N., Ermine, J.L., Aubertin, G., Trivin, J.Y., "Knowledge Capitalization with a knowledge engineering approach: the MASK method", In *Proceedings of IJCAI'2001 Workshop on Knowledge Management and Organizational Memory*, Seattle, USA, 2001.

McAdam, R. Reid, R., "SME and large organization perceptions of knowledge management: Comparisons and contrast", *Journal of Knowledge Management*, Vol. 5, No. 3, 2001, pp. 231-241.

McMahon, C., Browne, J., *Cadcam, principles practice and manufacturing management*, Addison Wesley Longman, Harlow, 1998.

Michellone, G. Zollo, G., "Competencies management in knowledge-based firms", *Journal of Manufacturing Technology and Management*, Vol. 1, No. 1, 2000, pp. 20-41.

Mori, G., Paterno, F., Santoro, C., "CTTE: Support for Developing and Analyzing Task Models for Interactive System Design", *IEEE Transactions On Software Engineering*, Vol. 28, No. 8, 2002, pp. 797-813.

Nonaka, I., Takeuchi, H., *The knowledge creating company: how Japanese companies create the dynamics of innovation*, Oxford University Press, New York, 1995.

Penalva, J. M., "SAGACE method: the modelling of human designed systems", In *Proceedings of COMETT'93,* Rome, Italy, 1993.

Polanyi, M., *Tacit Dimension*, Doubleday & Co, New York, 1966.

Rasovska, I., Chebel-Morello, B., Zerhouni, N., "A mix method of knowledge capitalization in maintenance", *Journal of Intelligent Manufacturing*, Vol. 19, No. 3, 2008, pp. 347-359.

Rikowski, R., "Value - the Life Blood of Capitalism: knowledge is the current key", *Policy Futures in Education*, Vol. 1, No. 1, 2003, pp. 160-178.

Sallaou, M., Taxonomie des connaissances et exploitation en conception preliminaire - Application à un systeme eolien, Thèse de Doctorat, Arts et Métiers ParisTech, 2008.

Savransky, S.D., *Engineering of creativity, Introduction to TRIZ methodology of inventive problem solving*, CRC Press, Boca Raton, 2000.

Schreiber, A.T., Akkermans, J.M., Anjewierden, A.A., De Hoog, R., Shadbolt, N. R., Van de Velde, W., Wielinga, B.J., *Knowledge Engineering and Management: The CommonKADS Methodology*. MIT Press, Cambridge, 1999.

Shahin, T.M.M., Andrews, P.T.J. , Sivaloganathan, S., "A design reuse system", In *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* , Vol. 213, No. 6, 1999, pp. 621-627.

Steels, L., "Corporate knowledge management", In *Proceedings of ISMICK'93*, Compiegne, France, 1993a , pp. 9-30.

Steels, L., "The componential framework and its role in reusability", In J.M. David et al. (Ed.), *Second Generation Expert Systems,* Springer-Verlag, Berlin, 1993b, pp. 273-298.

Stone, R.B., Wood, K.L., Crawford, R.H., "A heuristic method for identifying modules for product architectures", *Design Studies*, Vol. 21, 2000, pp. 5-31.

Szykman, S., Racz, J.W., Sriram, R.D., "The representation of function in computer-based design", In *Proceedings of the ASME Design Engineering Technical Conferences*, Las Vegas, USA, 1999.

Tan, H.T., Libby, R., "Tacit Managerial versus Technical Knowledge As Determinants of Audit Expertise in the Field", *Journal of Accounting Research*, Vol. 35, No. 1, 1997, pp. 97-113.

Uluoğlu, B., "Design knowledge Communicated in Studio Critiques", *Design Studies*, Vol. 21, No. 1, 2000, pp. 33-58.

Wong, K.Y., Aspinwall, E., "Characterizing knowledge management in the small business environment", *Journal of Knowledge Management,* Vol. 8, No. 3, 2004, pp. 44-61.