INTERNATIONAL DESIGN CONFERENCE - DESIGN 2010
Dubrovnik - Croatia, May 17 - 20, 2010.

BOTTOM-UP KNOWLEDGE SHARING IN PSS DESIGN. A CLASSIFICATION FRAMEWORK

M. Bertoni

Keywords: product service systems, functional product development, knowledge engineering, knowledge sharing

1. Introduction

The emerging globalisation and the increasing market competitiveness are pushing companies to reconsider their business model, shifting from offering merely physical artefacts to introducing product-service combinations able to satisfy increasingly sophisticated customer needs. The basic principle of the Product Service Systems (PSS) paradigm is to bring added value to the customer by offering the "functionality" or "performance" of the physical product as a mix of goods and service. Hence, the risk related to maintenance and availability of the "product" lies with the manufacturer, adding an incentive for developers and providers to explore ways of designing it as effectively as possible from a life cycle perspective, to maximize the profit and value that can be drawn from it.

This aspect radically changes the scope and objectives of the engineering activity [Larsson 2008]. The need to raise the knowledge baseline for new product development projects, integrating many diverse knowledge areas (e.g. engineering design, manufacturing, business development, etc.) becomes even more evident.

An enhanced understanding of the full range of lifecycle demands and needs (e.g. related to in-use behaviours, expected maintenance cycles or recycling considerations) has to be reached at the earliest possible time, to better understand the function to be developed. In such a situation, the availability of "downstream" knowledge, i.e. from the later life-cycle phases, becomes crucial to improve early-stage decision-making.

In a "traditional" product development situation, the embodiment of product properties into 3D models and data structures provide a good enough basis for the decision-making activity. Hence knowledge sharing is well supported by computer applications such as CAD (Computer Aided Design), CAE (Computer Aided Engineering), PDM (Product Data Management), and PLM (Product Lifecycle Management). Engineering work, however, also includes a social dimension, e.g., learning and experiencing, i.e. what engineers know by practice deeply influences product design decisions. This is particularly evident in PSS design, where the intangible aspects of the product come strongly into play and knowledge is usually tacit, not made visible in the same way as technical product information. Moreover, a single company rarely has all the knowledge required to deal with the development and supply of a mixed offer, therefore companies typically team up in partnerships of an Extended Enterprise (EE) or Virtual Enterprise (VE) kind. Hence engineers and designers are increasingly requested to work in highly cross-functional, cross-disciplinary, enterprise-wide teams and to develop closer interactions with external stakeholders to gain a better understanding of how value can be generated for the customer and the final user.

In such a context "traditional" top-down CAx/PDM/PLM systems, characterized by hierarchical structures that progresses from a large, basic unit to smaller, detailed subunits, show limitations in the way knowledge identification, capturing and sharing is supported. The recent development of social computing and Web 2.0 suggests that a more “unstructured”, “self-governing” and bottom-up approach (i.e. Engineering 2.0 [Larsson 2008]) could potentially address some of the shortcomings of the “traditional” solutions.

2. Motivation and objectives

In recent times it has been observed an increasing trend towards the adoption of Web 2.0 technologies in industry and specifically in a product development setting. A recent report from Gartner Inc. [Mann 2009] foresees that the 80% of enterprise collaboration platforms by 2013 will primarily be based on 2.0 techniques. Other authors have discussed the impact of bottom-up and lightweight tools in knowledge management [Richards 2009], sometime using the term Knowledge 2.0 to summarize the upcoming trend [Levy 2009]. Eventually, McKinsey [McKinsey 2009] outlines that 2/3 of the companies interviewed during 2008 on the topic of Web 2.0 are using or are eager to implement these tools in the development of products and services.

Moving from this ground, the purpose of the research is to investigate the possibilities offered by bottom-up and lightweight technologies to support cross-functional knowledge sharing in a PSS context. The purpose of this paper is to analyze the top-down vs. bottom-up dichotomy [Shapiro 1987] under a PSS perspective and to explore how a collective sense-making knowledge sharing approach could potentially leverage the team performances in the design of product-service combinations, by defining a list of dimensions to benchmark Web 2.0 tool capabilities in the design of products-services.

These parameters constitute the backbone of a framework aiming to raise the awareness of design stakeholders on the possibilities offered by a lightweight and bottom-up technologies. The framework is intended as a means to communicate, in an easy and intuitive way, how tools like weblogs, wikis or tags could be beneficial for design teams in a cross-functional environment, underlining similarities and differences in a meaningful way for engineers. Eventually, the framework aims to raise the awareness on the capabilities offered by technology mash-ups, visualizing of how knowledge sharing problems could be addressed combining different 2.0 tools.

3. Method

This paper summarizes the intermediate results of a project conducted in collaboration with Swedish manufacturing companies in the topic of cross-functional knowledge sharing. The framework proposed draws on data from several development projects related to products in various industry segments, ranging from the development of manufacturing tools to aircraft engine components. For the purpose of this research, the design research methodology framework (DRM) [Blessing and Chakrabarti 2009] has been adopted. The DRM framework consists of four main phases: *Research Clarification*, *Descriptive Study I*, *Prescriptive Study*, and *Descriptive Study II*. The framework here presented is a main outcome of the *Descriptive Study I* step.

The research strategy combines a case study approach and ethnographic methods to verify the approach close to technology and product development activities in industry, accessing empirical cases. Several multi-day workshops, virtual meetings, and company visits have been performed during the data collection phase. Semi-structured interviews have been initially conducted with the scope of picturing the State-of-Practice in industry, while the detailed data gathering phase in-situ observations, group interviews, in-depth interviews have coplemented the analysis of working documents. Overall, the research has involved about 30 people with knowledge on these projects, both from academia and from industry.

4. Top-down vs. bottom-up knowledge sharing in a PSS context

In many engineering situations, the information and knowledge embodied into 3D product models and data structures provide a good enough basis to support decision-making activities. Engineering work, in fact, relies on expertise and competences that relate to the product that is being developed, i.e. the physical thing, thus decision making activities are commonly well supported by computer applications such as CAD (Computer Aided Design), CAE (Computer Aided Engineering), PDM (Product Data Management), PLM (Product Lifecycle Management) and KBE (Knowledge Based engineering).

A closer look shows that all these systems show prevalent *top-down* [Shapiro 1987] features. They emphasize the planning and the complete understanding of the system to be developed (e.g. a vehicle, a screwdriver, a printer, etc.) to categorize the different product models. Once a commonly agree view of the system properties is established, the system is cascaded down in its constituent parts until a base element is reached. Sub-systems and components are arranged in a hierarchical structure and linked by a generalization-specialization relationship. Hence the knowledge elements attached to them are classified according to the established taxonomy.

The definition and maintenace of such taxonomy requires experts' involvement to ensure that a coherent reference for all the stakeholders is maintained. As far as the top-down structure is populated, data, information and models are further interpreted to fit changing situations and perspectives. What makes sense in one context can change or even lose its meaning when communicated to people in a different context, thus generating confusion, ineffective knowledge categorization and sharing. Experts need to keep the control over the system, which it becomes highly dependent from their active contribution.

In a situation where the purpose is to develop a "function" instead of a physical product, no solutions are evident in the early design stages, intuitively because a function can be obtained by different combinations of products and services. At this step, designers are requested to abstract their reasoning and elaborate solutions from an overall system perspective, gathering knowledge all over the place and relating it to the top-level PSS architecture. The lack of a well established reference model in the early beginning makes difficult to define a commonly agreed taxonomy until later in the process. Moreover, downstream knowledge assets cannot be easilty expressed as system or sub-system properties, especially in case they relate to a "soft" and "intangible" dimension, such as service issues, customer opinions or feelings.

In a Virtual Enterprise environment, made of autonomous and independent players, major companies can rarely impose their preferred mode of collaboration, technology, or even agree on some sort of standard. VEs are also highly volatile partnerships, built on a project-by-project base, that are disbanded when the business opportunity has passed. The high level of uncertainty that characterize the agreement may discourage the idea of allocating precious resources for the development of an ad-hoc system and categorization framework. Eventually, it may be difficult to identify experts that have knowledge on all the aspects of the product, thus able to establish a coherent taxonomy and to supervise the knowledge sharing system as long as it is populated.

In synthesis, the use of traditional top-down systems in PSS design may to some extent inhibit the effective flow of knowledge, leading to situation in which design decision are taken on the basis of a far-than-optimal knowledge base. Imposing a rigid categorizations framework may hide the relationships between knowledge assets in different domains, thus making difficult for the engineers to generate radically innovative ideas.

All these aspect makes particularly interesting to investigate the bottom-up paradigm in relation to PSS design. In a bottom-up model, in fact, the agents of knowledge creation are all the single individuals, not merely the top managers or the system experts [Nonaka 1988]. The information is organized not according to hierarchical rules, rather following a self-organizing principle. Leaders and experts do not try to prevent chaos and noise by imposing their control over the system, rather chaos and noise are permitted, and they act as sponsors for the knowledge creation activity [Nonaka 1988]. Hence, the system dependency from the experts is reduced and the focus is shifted from explicit, computerized and documented knowledge to the tacit knowledge incarnated in individuals.

Bottom-up knowledge sharing represents an opportunity to make structural knowledge capital out of the human knowledge capital and to tap into such "wisdom of crowds", helping engineers to keep their social ties loose, to keep themselves exposed to as many diverse sources as possible, and to participate in groups that range across hierarchies [Surowiecki 2004]. It has to be noted, however, that the purpose is not to replace top-down systems, rather to complement them to answer different questions, thus improving the capability to capture, model, simulate and share knowledge across organizational and departmental boundaries.

5. Previous work

Bottom-up knowledge sharing is not really a new concept and many of the technologies/methods have already been discussed and established. What is interesting to investigate here is the possibility to use these technologies for the benefit of engineers and stakeholders involved in product-service development.

The Web 2.0 idea, which is revolutionizing the World Wide Web in favour of a more collective sense-making approach to information and knowledge exchange, is made of tools such as Weblogs, Wikis, Social Networking, Tagging, RSS, Mashups, Podcasts, Bookmarking, Media Sharing, Collaborative Editing, etc. The Future Exploration Network (http://futureexploration.net/) has attempted to map this wide range of applications across two major dimensions: Content Sharing to Recommendations&Filtering and Web Application to Social Network. The four spaces that emerge at the junctions of these dimensions are Aggregation/Recombination, Widget/Component, Rating/Tagging, and Collaborative filtering. Collectively these dimensions cover the primary landscape of Web 2.0.

Many organizations have started to use Web 2.0 technologies in their working environment. MacAfee summarizes the rising company interest on the use of 2.0 tools for generating, sharing and refining knowledge in a global setting, with the term Enterprise 2.0 [McAfee 2006]. He proposes a framework named SLATES to specify the six underlying components of Enterprise 2.0 technologies: Search, Links, Authoring, Tags, Extensions and Signals. The FLATNESSES framework, later proposed by Hinchcliffe, adds four dimensions to SLATES and to the Enterprise 2.0 definition: Freeform, Network-oriented, Social, and Emergence.

Although these frameworks provide useful indications regarding the main characteristics of the Web 2.0 trend, they poorly address the knowledge sharing issue, neither they explicitly target engineers and engineering work, nor they refer to PSS design.

The work has been focused, therefore, on the definition of a categorization framework able to describe the bottom-up technology capabilities from a PSS viewpoint. Such a classification approach intends to facilitate the stakeholders in benchmarking methods and tools and in identifying the ones able to cope with their specific knowledge sharing problems. Since modularity and flexibility are main characteristics of 2.0 technologies, the framework aims also to show to what extent the product development activity may benefit from the integration of different applications, i.e. from Web 2.0 mash-ups.

6. Framework definition

One of the key success factors in the development of product-service offerings in a Virtual Enterprise setting is the ability to effectively and seamlessly assemble and utilize, drawing from the different perspectives of the involved partners, the individuals' pool of resources and the various combinations of their specific capabilities [Larsson 2008]. Optimal solutions are not likely to be found in the engineering area alone, as the function can be provided both as a product, as a service or a mix of them. This asks for a more explorative approach that could provide relevant stimuli for the design activity.

In a PSS situation, the providers maintain the ownership of the product and become responsible for the availability of the function in front of the customer. Deepening the understanding on the customer usage behaviours would potentially suggest innovative and more optimal product concepts. One of our informants in the engine components manufacturing business has pointed out that knowing how pilots operates the engine during take-off and landing may suggest to completely rethink the architecture of the product to make it more profitable under a lifecycle perspective. In an extreme scenario, it might be the case of using ABS instead of exotic composite materials for components that are expected to be inspected on a regular basis. ABS components, although not offering the same performances in terms of stress resistance, are cheap and might be simply replaced at each inspection. They might also allow introducing expensive features at reasonable cost, thus changing the cost structure. This knowledge, however, is difficult to capture and store in the "traditional" CAD/PDM/PLM systems and is not easily visible As-Is to the design team in an early phase. The capability to stumble upon' relevant inputs for the design activity, such as knowledge that has been accumulated in previous projects or that relates to other domains and disciplines, is critical in product-service design. Serendipity [Larsson 2008] summarize the system capabilities to elicit, codify, structure and make available knowledge from different sources to the engineers in an early design phase.

A drawback is that the “noise” of the search increases proportionally with the number of sources and elements browsed during the search. Being able to find the right information in the shortest possible time is therefore crucial to reduce the process lead-time. A key capability for the tool relate to precision, i.e. the number of relevant hits over the number of elements retrieved.

In PSS design, inputs may be found in many different places, teams, functions or companies, ending in a situation where plenty of time is spent just to access different knowledge sources and to browse through the knowledge elements. For this reason, the knowledge sharing tools should support knowledge aggregation, i.e. the combinatins of different elements on the same place/screen/page, visually merging information in a way that is relevant for the final user, thus saving people’s time by speedign up the search activity across different domains.

Then, the identification of relevant information is also dependant by the level to which knowledge elements are linked one each other, thus by the possibility to quickly browse a wide variety of topics that makes sense to others.

A key challenge when shortening development cycles is ensuring that decisions can be taken earlier, without decreasing the effectiveness of the decision. One of our informants in the tool manufacturing industry has brought up the following example to explain common problems related to the management and reuse of product-service knowledge in a cross functional situation:

“….Lasse is an experienced product development engineer working on new cutting insert design. In order to sort out good and bad knowledge elements for later reuse, he is used to copy on his local drive the files contained in the different project repositories, renaming them by adding a (+) in front of the file name if the content is particularly interesting for the project purpose, a (=) if the description may be useful to some extent for the development of the new tool, or (-) if it does not provide any crucial knowledge to fulfill the project goals.”

To make decisions quicker, decision makers need to improve their understanding of the ”fitness-for-purpose” of a piece of information or knowledge, to ensure that the decision making process is guided by well-founded ideas, principles, theories, insights. This is particularly evident in a PSS context, where knowledge relates both to tangible and intangible domains, and is characterized by different levels of granularity. A crucial capability for a knowledge-sharing tool is to provide quick feedback regarding the applicability of a certain piece of knowledge in a design situation.

While acknowledging that new product development and innovation will always be an exploratory activity, involving high levels of ambiguity and uncertainty, it becomes more important than ever before to take decisions with appropriate consideration of the maturity or readiness in relation to the decision making objectives [Johansson 2009]. A key capability for a knowledge sharing tool is to increase decision makers’ awareness and understanding of the current status of the knowledge base on which they draw upon when making decisions. In particular, cross-boundary discussions focused on the perceived maturity of available knowledge assets need to be supported, to make sure that any potential risks related to limitations in the knowledge base can be identified and mitigated.

Recent analysis show that more than 80% of the organizational data are unstructured [Bell 2006, p.271] and these figures are likely to be higher in a PSS situation, due not only to the increased “intangibility” and “ambiguity” of the problem statement, bu also to the high turnover rate that characterize the Virtual Enterprise. Minimizing person the dependence of the knowledge baseline and helping newcomers in exploiting the knowledge of more experienced engineers is a main objective here, and it can be achieved by developing tools able to locate expertise [Larsson 2008] in the organization. This problem is further complicated by the fact that the foremost experts might not even be on the company payroll and there might be “hidden experts” around the enterprise, outside the official job description, as well as lead-users in any part of the customer network, offering their advise and experience.

Dealing with the provision of functions, it is important to store the decisions related to options that has been rejected and the reasons for not following specific paths, because conditions might change character in new contexts where similar situations have to be evaluated. A relevant dimension is the capability to capture the rationale behind a design decisions, making visible the underlying argumentations that justify why a design option seemed to be/not to be feasible within the context in which the decision was taken.

Studies in decision making have shown that the efficiency of a decision made has an inverted U shaped relationship with the amount of information provided. A synthesis assessment is often preferred to a number of unsorted information. In a PSS context this problem is exacerbated by the huge amount of inputs to be handled, thus the capability to provide only the "golden nuggets" to the decision makers is an important aspect on which to evaluate the tools. Synthesis is main issue also when searching for relevant knowledge. Designers are requested to browse and retrieve information from a number of different sources, thus the possibility to provide a synthesis assessment on the content of a given repository would be highly appreciated to reduce the time for the search.

In the later design process phases, PSS engineers are also requested to deal with very domain-specific issues. The right answers, or the right people to ask to, are even more difficult to locate than in a traditional engineering situation. Questions may be so specific that only a few people in the entire organization may be have relevant knowledge. Supporting engineers in finding specific answers to their questions is, therefore, an important dimension to be considered when working in a PSS design situation.

The availability of up-to-date knowledge is crucial to coordinate the efforts of the different teams and to reduce the risk of taking decision on an incomplete basis. One of our informants in the cutting tool industry, reasoning on the development of product-service combinations, has outlined that:

"…when you look at the customers, there are situations in which they have created the innovation by using the hardware in a certain way. They might use the product differently than we had thought during development, and this can be an opportunity for innovation if we get to know this in a reasonable time."

Knowledge is a "living" thing that, once captured, stored, shared and used, tends to become old and out-to-date over time. Therefore, knowledge sharing tools should be able to facilitate the *capturing* and integration of such updates in the knowledge base, to avoid ending in situations where the content of the system and the enhancement of it is separated from those who deal with the knowledge generation processes in their daily, regular work activities.

Due to the number of actors participating to the design activity, the problem is not merely related to the capturing of the updates and to their integration in the existing knowledge base. Several of our informants have pointed out that it is also related to the possibility of *pushing* such updates to the right people as far as the work progresses, to ensure that every design activity and every decision is based on the best available knowledge.

6.1 Defining a scale

A single scalar from 0 to 5 may be assigned to each 2.0 tool against each dimension identified above:

- 0 is used when the dimension is not applicable to a given tool (e.g. RSS feeds are not intended as a tool to formalize the design rationale).
- 1 indicates tools that might support a given task (i.e. push updates) but that are not intuitive to use, requires great engagement and superior coordination (e.g. forums might provide room for collecting the argumentations behind a given design decision, but may result cumbersome to use).
- 3 indicates tools with good capabilities (and some limitations) in a given area (e.g. the discussion in the wiki may facilitate the identifications of experts around a topic, although accurate information about their real competence and expertise may be lacking).
- 5 indicates excellent capabilities (e.g. tags may allow retrieving documents that relates to different functions, disciplines or databases).

The meaning of 1, 3 and 5 for each dimension is described in Table 1:

Table 1. Scales in the framework

SERENDIPITY	5 – Knowledge elements from heterogeneous sources are identified and retrieved. 3 – Knowledge elements in similar disciplines or fields are identified and retrieved. 1 – Knowledge elements in the same source are identified and retrieved.
PRECISION	5 – The rate of relevant vs. retrieved knowledge elements is high. 3 – The rate of relevant vs. retrieved knowledge elements is medium-low. 1 – The rate of relevant vs. retrieved knowledge elements is very low.
AGGREGATE	5 – The tool aggregates knowledge elements from heterogeneous sources. 3 – The tool aggregates knowledge elements from the same source. 1 – Knowledge elements are displayed once at a time.
LINK	5 – The tool allow establishing links to knowledge elements across functions. 3 – The tool allow establishing links to knowledge elements in the same area/discipline. 1 – Very few links can be established between the knowledge elements.
APPLICABILITY	5 – Complete information on the applicability of a knowledge element is given. 3 – Limited information on the applicability of a knowledge element is given. 1 – Very poor information on the applicability of a knowledge element.
MATURITY	5 – Complete information to support knowledge maturity assessment is given. 3 – Limited information to support knowledge maturity assessment is given. 1 – Very poor information to support knowledge maturity assessment is given.
EXPERTISE	5 – Best practices, preferences, patterns and working approach are captured and recorded. 3 – Personal opinions, comments and connections are captured and recorded. 1 – Comments are captured and recorded.
RATIONALE	5 – Complete information regarding the rationale behind a certain decision is given. 3 – Limited information regarding the rationale behind a certain decision is given. 1 – Very poor information about the rationale is given.
SYNTHESIS	5 – Suitable to provide a synthesis assessment of a document/knowledge source. 3 – Suitable to provide a quick overview of the content of a document/knowledge source. 1 – A synthesis assessment cannot be automatically derived.
SPECIFICITY	5 – Suitable to capture highly problem- and process-specific knowledge. 3 – Suitable to capture domain-specific knowledge. 1 – Suitable to capture generic knowledge and contextual information.
CAPTURE	5 – The latest updates are recorded and made available almost real time. 3 – The latest updates are recorded and made available with delay. 1 – Updates are captured and made available with large delay.
PUSH	5 – Updates are forwarded real time to potential stakeholders. 3 – Updates are forwarded with delay to the potential stakeholders. 1 – Updates have to be manually retrieved by the stakeholders.

7. Application example

Figure 1 exemplifies the use of the framework, by comparing two common Web 2.0 tools such as weblogs and tags. Blogs might be used as a platform for early feedback from external stakeholders and employees and to nurture discussions on product and service offerings, lowering the threshold for commenting and documenting personal experiences. New ideas and findings (on innovation projects, on customer visits, etc.) could be captured and presented to a larger audience as entries in the weblog. With their comments, the blog users may also give some indications about the applicability of a piece of knowledge in their own (or even other) contexts. The two-way communication channel may also help in identifying experts on a given topic and to establish with them a continuous dialogue.

Using tagging practices, knowledge elements are put into different bins at a time (per customers, competitors, projects, product types, maintenance and service offerings) making easier for others to locate and fetch from different sources, thus facilitating the serendipitous discovery of knowledge that would have been normally been considered in isolation. The list of tags associated to an element may also give info, to an expert eye quick, about the applicability of a knowledge element, without necessarily reading a full report or deliverable. Similarly, tag clouds may give a preview of the content of a weblog or a repository. Less intuitively, tagging may be used to explicit the maturity associated to a piece of knowledge, i.e. tagging an element according to its current in the knowledge maturity scale. Attaching or detaching tags might be a quick and easy way to capture updates and changes related to an existing piece of knowledge, although it requires great coordinations between the teams. Specific tags can be also attached to elements to suggest tasks or situations in which the knowledge has grown or in which it might be applicable. Eventually, they may help to find people who tags things the same as a designer does, suggesting people with the same interests, experience or tasks in the virtual organization.

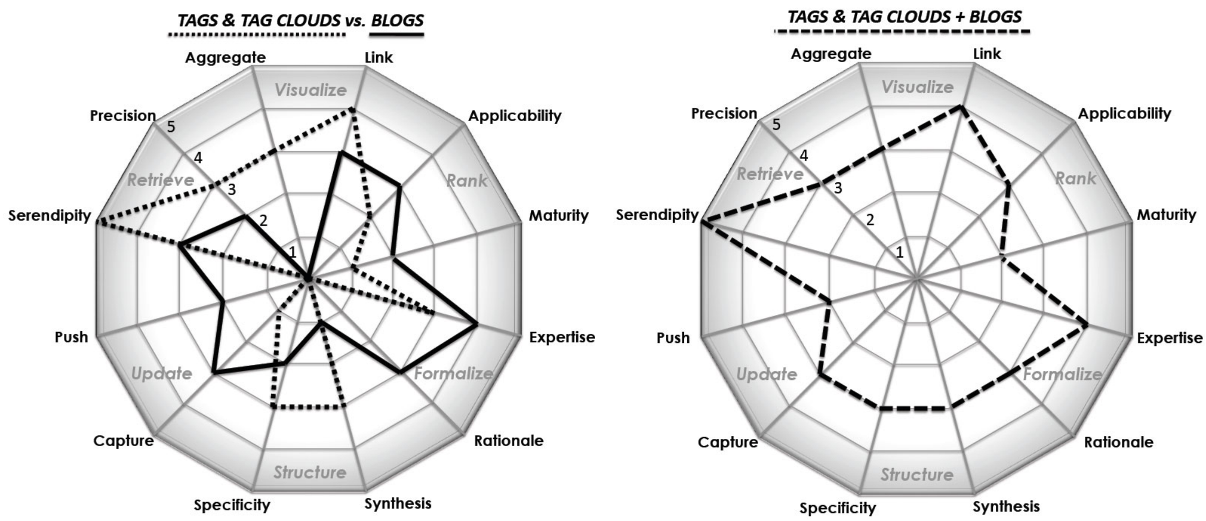

Figure 1. Tags and blogs, comparison and integration

Figure 1 shows a benchmarking (left) and integration (right) example. Weblogs (continuous line) and tags (thin-dashed line) have been initially compared to analyze their features from a PSS perspective. Then, an example of how the tools could address the knowledge challenge by developing a mash-up is given on the right (dashed line). The mash-up capabilities are obtained by interpolating the points with the highest value for each parameter. The resulting contour is intended to show an ideal situation where the tools are combined in an optimal way, outlining go and no-go areas.

Since bottom-up tool effectiveness is strictly related to the user participation in the knowledge sharing process, the technologies are evaluated considering users characterized by medium-high participation. Participation, however, is not the only variable that may influence the technology effectiveness. User skills and the availability of best practices also influence the performances of the tools. In the framework proposed, users are assumed to possess medium-high skills.

8. Discussion

The framework presented in the previous section has been applied in the conceptual design phase of a Web 2.0-based knowledge sharing demonstrator, which is currently under development. The application of the visual approach proposed has shown to be successful in raising the industrial partners' understanding on the capabilities of 2.0 applications and to facilitate the discussion regarding features integration and mash-up development.

The discussion with the companies has also outlined some weaknesses in its definition. On one hand, the framework does not fully describe the complexity of the knowledge problem in a PSS context and neglects some parameters considered crucial by the industrial partners, such as privacy and security. The mode of coopetition that characterize VE agreements, in fact, further raise the fear of losing control over the knowledge flows and to suffer from leakage of proprietary know-how. This is an important aspect to deeply consider in the developing of a bottom-up approach in this context. This also outlines the difficulty of defining parameters able to map a wide range of Web 2.0 applications across dimensions considered relevant by PSS stakeholders. In this research 12 major aspects have been chosen, and these might be cascaded down sub-dimensions. *Maturity*, for instance may decomposed into dimensions like *inputs*, *methods* and *expertise* [Johansson 2009]. This may change considerably how tools are combined to reach superior capabilities with a mash-up. Finding the right trade-off between the number of dimensions to be displayed may not be easy and depends on the industrial sector considered and on the characteristics of the PSS offer (e.g. product-oriented instead of result-oriented).

On the other hand, the value scale proposed for each parameter may be questionable. Although the intention is to qualitatively assess the technology capabilities, a more defined and fine-grained scale might be needed. Moreover, mapping the process and tools in the framework may not be that straightforward, as the value associated to each dimension may be subject to interpretation. A larger area in the diagram does not necessarily identify the best option to cope with a given problem, rather it suggests a solution direction that needs to be further analyzed with critical eyes and contextualized.

One of the main problems in the evaluation of bottom-up tools is related to the effective participation of the users. Participation may be inhibited by several factors such as technologies being too time consuming, information not being accurate or relevant for people or tools too difficult to use. The above-mentioned capabilities are strongly dependent upon the operator capabilities to unfold knowledge from what is seen in the tool. The availability of guidelines is important, since the effective use of the technology is not only dependant by the specific skills, ability and motivation of the individuals, but also by the way the are supported in learning and using the tools (such as the availability of best practices or ad-hoc training). People might use tags a lot and may still be bad taggers, reducing the potential benefits associated to the tool. Even the most intuitive technique fails to provide any benefit if not adequately supported in terms of methods and guidelines.

9. Conclusions

This paper presents an approach to visually capture the knowledge sharing capabilities of bottom-up and Web 2.0-based tools in the design of Product Service Systems. The main result of this work is a visual approach to communicate how such technologies may be beneficial for engineers working in the design of product-service combinations. Differently from existing Web 2.0 categorization frameworks, the approach focuses on dimensions that relate to practical issues engineers are dealing with when working in cross-functional teams and in a Virtual Enterprise environment.

First, a set of parameters relevant within a PSS development context is proposed. They have been defined in close collaboration with industrial companies and accessing empirical cases. For each parameter, a scale is defined to measure the tool capabilities with PSS lenses on. A visual framework is eventually proposed to map the different bottom-up technologies and to visualize the knowledge sharing potential of 2.0 mash-ups.

The visual framework proposed has been used in the early development phase of a Web2 2.0 prototype, showing to enhance the common understanding of the capabilities of bottom-up and lightweight tools in product development and to facilitate the discussion among the requirements for the lightweight demonstrator.

In the current formulation, the framework is not suited for a quantitative evaluation of bottom-up technologies capabilities. The purpose is merely to describe the aim of the tools, rather than to define a metrics for measuring their effectiveness. However, its ultimate scope is to support the development of knowledge sharing indicators able to measure the effective usage of the technologies in a real working design situation, i.e. evaluating how much knowledge has been serendipitous discovered or how much knowledge has been found to be applicable for a certain problem in a given timeframe. Further work will focus on the definition of such metrics on the basis of the proposed parameters.

From a process perspective, the company have expressed a great interest towards the possibility of using bottom-up knowledge sharing technologies not only to support the classical *recognizing symptoms-implementing corrective actions* mode. Backtracking the causes for a failure is just one part of the job. Helping designers in preventing mistakes, thus supporting a root-cause analysis and simulation-driven approach is seen by the industrial partners as a main opportunity with this approach.

Further research will focus on how this working mode could be supported by bottom-up methods and tools.

Acknowledgements

We sincerely acknowledge the Faste Laboratory, a 10-year Swedish VINNOVA VINN Excellence Center focused on Functional Product Innovation.

References

Bell, S., "Lean enterprise systems: using IT for continuous improvement", John Wiley & Sons Hoboken NJ, 2006.

Blessing, L.T.M. and Chakrabarti, A., "DRM, a Design Research Methodology", Springer Verlag, London, 2009.

Johansson, C., "Knowledge Maturity as Decision Support in Stage-Gate Product Development: a Case from the Aerospace Industry", PhD Thesis, Luleå University of Technology, 2009.

Larsson, A., Ericson, Å., Larsson, T., Randall, D., 2008, Engineering 2.0: Exploring Lightweight Technologies for the Virtual Enterprise, Proceedings of the 8th International Conference on the Design of Cooperative Systems, COOP 08, Carry-le-Rouet, France, 2008.

Levy, M., "Web 2.0 Implications on Knowledge Management", Journal of Knowledge Management, Vol.13, No.1, 2009, pp 120-134.

Mann, J., "Wikis and documents represent different ways of working", A Gartner white paper, 2009.

McAfee, A., "Enterprise 2.0: The Dawn of Emergent Collaboration", MIT Sloan Management Review, Vol.47, No.3, 2006, pp. 21-28.

McKinsey, "Building the Web 2.0 Enterprise: McKinsey Global Survey Results", McKinsey global survey, 2008.

Nonaka I., "Toward Middle-Up-Down Management: Accelerating Information Creation," Sloan Management Review, Vol.29, No.3, 1988, pp 9-18.

Richards, D., "A Social Software/Web 2.0 Approach to Collaborative Knowledge Engineering", International Journal of Information Sciences, Vol.179, 2009, pp 2515-2523.

Shapiro, S.C., 1987. Processing, bottom-up and top-down. In Stuart C. Shapiro, editor, Encyclopedia of Artificial Intelligence. John Wiley & Sons, New York, 1987, pp 779–785.

Surowiecki, J., "The Wisdom of Crowds: Why the Many Are Smarter Than the Few and How Collective Wisdom Shapes Business", Economies, Societies and Nations, Doubleday Books, New York, 2004.

Marco Bertoni, PhD
Assistant Professor
Luleå University of Technology, Division of Functional Product Development
SE-97187, Luleå, Sweden
Telephone: +46 (0)920 492583
Telefax: +46 (0)920- 49 13 99
Email: marco.bertoni@LTU.SE
URL: http://www.ltu.se/tfm/fpd