Combining BPM and Social Software: Contradiction or Chance?

Selim Erol, Vienna University of Economics and Business Administration, Vienna, Austria

Michael Granitzer, Know-Center Graz, Austria

Simone Happ, T-Systems Multimedia Solutions GmbH, Dresden, Germany

Sami Jantunen, Lappeenranta University of Technology, Finland

Ben Jennings, University College London, London, UK

Agnes Koschmider, Institute of Applied Informatics and Formal Description Methods
University Karlsruhe (TH), Germany

Selmin Nurcan[1,2], [1] CRI, University Paris 1 Panthéon Sorbonne
[,2] IAE de Paris Sorbonne Graduate Business School, France

Davide Rossi, Dipartimento di Scienze dell'InformazioneUniversità di Bologna – Italy

Rainer Schmidt, HTW-Aalen, Aalen, Germany

Paul Johannesson, Department of Computer and Systems Sciences, SU/KTH, Stockholm, Sweden

Abstract

Due to many success stories, social software received much attention in public and academia. Although it is widely used for business support, its relationship with business process management has not been analysed. The results of the workshop on Business Process Management and Social Software as part of the international conference on Business Process Management in Milano workshop show the manifold possibilities to join concepts from business process management and social software. Social software offers new possibilities for a more flexible design of business processes and provides a better integration of employees. Particularly knowledge intensive processes may profit from social software techniques. Also the modelling of business processes may profit from using social software techniques. Social software can also be used in the implementation and deployment phase of the business process lifecycle. Thus it is also possible to provide workflow support in a social software environment. The use of social software also requires new considerations about digital identity and reputation in business processes.

1 Introduction

Social software supports social interaction and social production and raises the level and scope of interaction facilitated by computer and computer networks [1]. Social production is the creation of artefacts, by combining the input from independent contributors without predetermining the way to do this, how it was done in Wikipedia. Although the success of social software is rather new, its roots can be traced in the 1960s and even 1940s [2] . Concepts such as Granovetter's weak ties [3], [4] foresaw the power of social interactions without knowing the possibility to implement them using software systems.

Impressive results are created without a central plan or organisation. Instead, social software uses a self-organisation and bottom-up approach where interaction is coordinated by the "collective intelligence" of

the individuals; the latter does not necessarily know each other and are a priori not organised in a hierarchy. Furthermore, social software follows a rather egalitarian approach; decisions are not made by small elites but by combining a multitude of inputs from different users. Also terms and taxonomies are developed collaboratively and not imposed by an expert or a group of experts. Thus, content from different contributors is gathered and aggregated continuously and becomes immediately visible and effective. The content created is under continuous assessment of all users. Every user may detect and correct flaws in the content; without using a formalised change procedure. Trust and reputation play a crucial role in the use of social software instead of authority granted by the top management. Based on reputation, hierarchic structures can be defined by content driven roles instead of organisational hierarchies and self-organisations can be used more widely.

Social software does not only consider content but also context valuable. Therefore, many types of social software also support creating context (usage) information for a physical or digital object in the form of tags, links or bookmarks. By capturing also the context of information, not only semantics but also the pragmatics of information can be represented. Many different types of content are possible such as text, (web) documents or multi-media. Three sub-types of context can be differentiated: Annotation, reputation and social links. Context can be expressed by different technical means such as text, links etc.

Annotation is information that helps to understand, find, and evaluate objects. These objects may be digital or real objects. *Reputation* substitutes trust in social software. As most users of social software do not know each other, it is necessary to provide reputation information to evaluate the validity of information. Finally, *social links* provide information about connection between human beings and establish social networks.

Social software is used by enterprises to support concepts such as Enterprise 2.0 [5]. It supports new communication patterns between customers and the enterprise. The former unidirectional communication from the enterprise to the customer is replaced by a multidirectional one. The customer may provide information to the enterprise using blogs to capture and share their ideas for new products and features. Customers communicate among themselves to support each other and to exchange knowledge about the beneficial use of the company's products.

However, it is not only possible to use social software for the support of business processes but also for the support of business process management. How to combine concepts from business process management and social software will be the theme of this paper. It will proceed as follows. First, current issues in the use of business process management will be analysed. Then, supports which can be provided by social software to business process (re)design will be presented. The next section will be devoted to concepts of social software which can be applied to the deployment and performance of business processes. Based on these considerations, a dimensional view of utilising social software within enterprises is given. Finally considerations about digital identity and reputation in business processes using social software are made.

2 Current Issues in the use of Business Process Management

There are two interrelated issues in the use of BPM where the use of social software could help.

Model-Reality Divide: On various conferences such as the International Conference on Business Process Management (BPM) or the Workshop series on Business Process Modelling, Development, and Support (BPMDS)[1] the so-called model-reality divide has been discussed. It is the divide between abstract process models and the executed processes. Thus, although business process models and structures are well designed, they are not used during the performance of business processes. Thus, the modelled and the executed (real) processes fall apart. Not surprisingly, the employees do not accept such business process models but "live" their own processes.

[1] For more details on the BPMDS series see [http://lamswww.epfl.ch/conference/bpmds08]

Lost innovation: Another important problem is the loss of innovations. Although there is knowledge in the organisation about possible improvements of business processes, this knowledge is not applied and the possible optimisations are omitted. Even further, the existence of such knowledge is unknown to the process owner.

The roots of the model-reality-divide and lost innovations are manifold and interconnected and shall be analysed know and are shown in the figure below.

Information pass-on threshold: Ideas for improvement are not passed on to the responsible because this creates too much effort for the latter and/or for the user ("Why shall I write a memo or a letter"). The further processing is not transparent to him ("What will happen with my suggestion...") or the success is considered as improbable ("Will not succeed anyway …"). The information pass-on-threshold may also be created when the entering of information is strongly regulated; the process to submit changes is too restrictive or simply takes too long due to approval steps. This causes that users cannot bring in easily their ideas. Therefore, important and valuable information is lost and improvements remain undone. As a consequence, the implemented process differs more and more from the best practice and the employees tend to do their own “private” process which contains the optimisations that they have regarded too difficult to integrate in the “official” process. Thus again a model-reality divide is generated.

Lack of information fusion: The other root of the model-reality divide and lost innovations is the lack of information fusion. Not all stakeholders and especially the users are properly involved into business process modelling. This may be caused by the exclusion of users by organisational means or by creating thresholds through the use of a formalised modelling tool. Thus the users are only "consumers" who are forced to accept the processes created for them. Also the inappropriate unification of terms by a top management driven approach instead of a peer-to-peer basis may create a model-reality-divide.

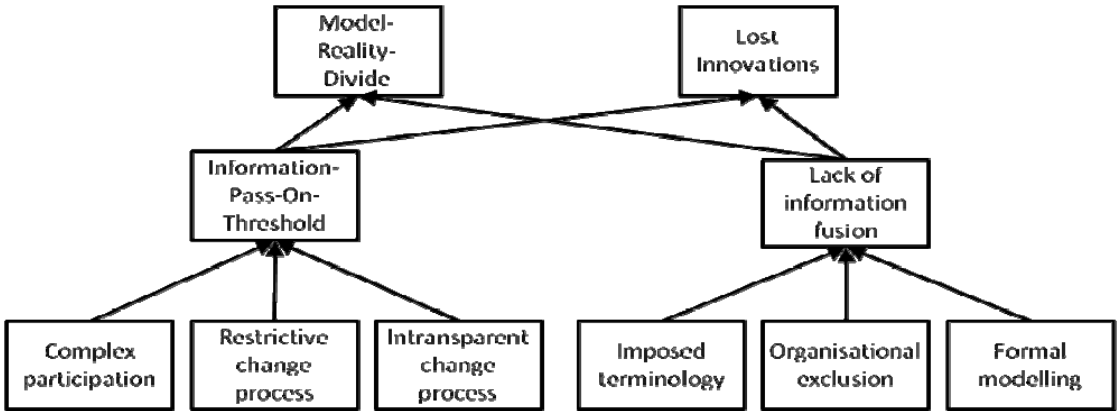

Figure 1: Roots of the model-reality-divide and lost innovations

It becomes easily obvious that social software allows coping with the roots of the model-reality-divide and lost innovations. The information-pass-on-threshold is lowered, because it is easy to participate in the design etc. of processes using social software. There is no complex change process. Instead, due to the change logs, all changes and their history are absolutely transparent. Also the other root of the model-reality-divide and lost innovations, the lack of information fusion can be coped with. Social software allows developing easily a common terminology, which may contain the understanding of terms of all participating users. Due to its egalitarian nature, an organisational exclusion can be avoided. Furthermore the use of natural language avoids the creation of participation thresholds due to the use of formal modelling tools.

3 New opportunities provided by social software for the design of business processes

3.1 Basic considerations for the design of business processes using social software

A process can be designed not only to take advantage of conventional business process management system (BPMS) support, but also the capabilities provided by social software.

One metaphor for contrasting a conventional BPMS and social software is the assembly line and the work station. At the assembly line, workers are placed at fixed positions to complete specific work on artefacts. When one worker has finished his task the artefact is moved to the next worker. Workers are specialised to do one thing, and ideally there should be no need to communicate with others to do a job well. The main purpose of an assembly line is to route an artefact among workers and thereby coordinating their work. Typically, the aim of this work organisation is to produce large quantities of standardised goods. At the work station, in contrast, the artefact is placed at a fixed position and different workers modify it according to the demands of a customer. There is an ordering between tasks but not as strict as at the assembly line. It is common that the work process is goal driven and that the goals are formulated as fulfilment of the artefact specification at a date. Communication among workers is frequent and spontaneous. Typically this kind of work organisation is used when a non-trivial customisation of an artefact is needed, when unknown solutions to problems must be found, or when precise ordering of activities cannot be established beforehand.

Design and management of business processes using BPMS benefit from using social software as customisation of produced goods becomes more frequent and the exceptions become the rule. Design and management of business processes using social software benefit from conventional BPMS support when production goals and dependencies between tasks to reach those goals can be formulated.

3.2 The influence of social production on the design of business processes

Traditional BPMS and social software can both address how to manage work activities [6]. But social software provides a number of new instruments:

- *Self identification*. Any actor who would like to contribute to an activity is allowed to do this, and s/he thereby identifies herself as competent for carrying out the activity.
- *Transparency*. All work results are openly available to anyone.
- *Signing*. All work activities are signed upon completion by the performing actor.
- *Open modification*. Anyone can modify contributions by other actors.
- *Logging*. All activities are logged to provide a history of work activities.
- *Discussion*. Comments on work results and suggestions for modifications can be discussed, and even directly linked to content pieces.
- *Banning*. Actors exhibiting inappropriate behaviour can be banned.

These instruments should be considered when designing business processes. That is, when designing a business process one should consider how it can be supported by a conventional BPMS in conjunction with the novel instruments provided by social software. For this purpose, a number of methodological guidelines are provided:

- *Design processes with a minimum of control flow*. Use the control flow mechanisms of BPMS primarily for the controlling management activities, and try to use social software mechanisms for most other management activities.
- *Embed processes in a social context*. In many BPMSs, users have a very limited view of the processes in which they participate, often only seeing an in-tray as the interface. Instead, users should be given

access to a wider context of the processes including information about other people that may contribute to the processes as well as histories of previous process executions.

- *Design for a low activity threshold.* In many process designs, work activities are large-grained meaning that carrying out an activity requires a substantial effort. Instead, most work activities should be designed so that they require only a small effort to complete. By reducing the activity threshold in this way, users are encouraged to participate in the processes.
- *Use honour points for rewards*. In most organisational processes, users carry out their activities because they are instructed to do so by their superior. In most social software, on the other and, participation is voluntary. A middle way is to make use of the notion of honour points, i.e. a participant receives credits in the form of honour points for activities she carries out and will be rewarded when she has obtained a certain amount of points. The reward may range from informal acknowledgements over monetary reimbursements to the formal fulfilment of organisational requirements.

In summary, conventional BPMS provide adequate support for many types of business processes. Introducing social software provides opportunities to design businesses in novel ways. The benefit of introducing social software in the design is most accentuated when the business processes concern production of non-standardised goods and services. There are also benefits when the process demands a high level of communication and collaboration among performing actors. However, these new ways of working may require considerable time for acceptance, e.g. many people may not be prepared to make their comments and changes visible immediately on a wiki. Therefore, there is a need for best practices, marketing efforts, and change management [7].

3.3 Automating Knowledge Transfer and Creation in Knowledge Intensive Business Processes

One critical factor in utilising technologies for organisational knowledge exchange is the well known knowledge sharing dilemma. Viewing it as a cost problem, employees try to minimize their cost for sharing knowledge with others while maximizing their benefits. While it seems that sharing knowledge on the Web has overcome this dilemma by its evolution into the Web 2.0 or the Social Web it is not perfectly clear whether and how it can be transferred into a corporate setting [8], [9]. Capturing not only knowledge bearing artefacts but organisational processes raises another barrier, especially in the case of knowledge intensive or weakly structured business processes: how can users express their work in an understandable and reusable way and how to share resources used in their daily work with a minimum on effort? For example users usually have hard times explaining step-by-step how they compiled a report or in what information sources they searched for getting the required background information. Besides, the immediate benefit of sharing their work in such a way is unclear and thus it is very likely that users will not engage in sharing.

In particular reviewing Web 2.0 success stories like Blogger, Flickr or Wikipedia reveals that technology has played a central role as enabler. Compared to the publishing of information via HTML in the early days of the Web, the lower entrance barrier of Web 2.0 technology significantly reduced the cost in sharing knowledge. By aiming at transferring the social software paradigm to business processes, one has therefore to think of means to reduce those costs of sharing successful and efficient process –execution- patterns as well as resources needed in such processes.

As outlined in [10], capturing the work context may be used to capture some patterns of weakly structured, knowledge intensive business processes executed in day-to-day businesses. At the heart of this aggregation systems (see [10]) lies the context detector. Key strokes, mouse moves, resources touched and application switches a user performs in her daily work are gathered and aggregated by the context detector. Based on this information and previously learned models machine learning techniques determine the current task a user is engaged in. Experiments show a satisfiable accuracy of 75% [11], [12] and questionnaires

undertaken in user experiments also confirmed a high degree of user acceptance for this task detection system.

Utilising such a work context detector produces tasks a user is engaged in and resources relevant for those tasks. Tasks are stored in a formalised manner as Resource Description Frameworks (RDF) graphs, aiming at sharing particular task patterns among employees. Understanding the group of employees as a community, the context detector provides a structure similar to folksonomies; a tripartite graph consisting of user, resource and task. In this task-folksonomy, tasks can be seen as substitute to tags in folksonomies: a personal, user generated description of resources. In difference to tags, those tasks are generated based on human trained models thereby avoiding problems in how to describe tasks with one single label. Similar to comparing folksonomies to ontologies, the task-folksonomy can be seen as an informal representation of process knowledge compared to the formal representation usually used in business processes management. Capturing the work context yields to such a task-folksonomy which – in analogy what folksonomies provide for artefacts - again yields to different supporting areas in task and resource sharing:

- *Sharing of tasks and resources among users*: Users may search for tasks of their colleagues or for resources used by their colleagues. By exploiting structural aspects of the task-folksonomy retrieval quality may be increased. Furthermore, by applying just-in-time retrieval paradigms [13], tasks and resources fitting to the current work context of a user may be provided automatically. Also, knowing other employees with similar tasks may yield to informal groups and informal discussions in sharing experiences on workflows outside of the digital world.
- *Enrichment of resources*: Resources often used together in particular tasks allow inferring that they share something common, at least at a statistical base. Finding similar resources is an important function in organising resources for most organisations. However, with the growing amount of resources, purely content based similarities hardly estimate the users' notion of resource similarity. Task-folksonomies may provide additional means to estimate similarities between resources more accurately by being independent of the resources' content and by creating relationships between resources based on the observed usage.
- *Collaborative creation of knowledge intensive ad-hoc processes*: Task-folksonomies provides rich grounds for business process engineers to analyse tasks employees are engaged in as well as possible dependencies among tasks. Having a huge task repository increases the success of applying statistical process mining methods like those outlined in [14].

Although the creation of tasks-folksonomies seems to be a fruitful approach for bootstrapping the sharing of tasks and resource among employees and more generally in overcoming the knowledge sharing dilemma, critical aspects are still open. Especially privacy is one major issue. In [10] a high user acceptance of the system could be achieved by allowing user to switch off the context detector and by refusing to publish specific tasks.

A second aspect is the quality of the task-folksonomy. Relying completely on automatic means may not yield to a satisfiable quality. User intervention and user feedback may be needed to increase quality to a higher level. However, automatic means may also bootstrap such behaviour since employees are seeing the benefit of sharing tasks with their colleagues. Thus, besides reducing the costs of knowledge sharing, automating knowledge transfer may also help to make benefits immediately visible.

3.4 Enhancing business process modelling using social software

To exploit advantages of social software services in BPM, a recommendation-based modelling support system has been enhanced with social network features. The core of the recommendation system [15] is to take into account a process builder's modelling context and the modelling history of a community of users, which suggests process model parts to the user that may help him achieving an individual modelling goal. For this, the modelling support system works on top of a repository, which stores business process models

(respectively parts) previously designed and stored by users from the same enterprise or from the same business branch. A process model part is defined as a logically coherent group of process elements belonging together (e.g. approval, billing or shipping).

Through the "social" extension [16], process builders can gain insight who already selected and reused specific process models. Such an extension should encourage user trust and participation by those users who are unskilled. To implement such an extension three kinds of social networks are used to this end, respectively a social network from a process model repository, a social network from a user history, and a social network from an insertion history.

(1) *A social network from a process model repository*: This social network provides an organisational view of business process models. To derive the social network from process models, three types of metrics defined in [17] are considered. They are transfer of work, subcontracting, and cooperation. Assume an organisation is defined by the two performers i and j. The transfer of work metric reveals the frequency of passing work from the performer i to the performer j. This metric is based on whether for the same case an activity executed by performer i is directly followed by an activity executed by performer j. The subcontracting metric counts the number of times performer j executes an activity in-between two activities executed by performer i. The cooperation metrics counts how frequently the two performers i and j participate in activities of the same models.

(2) *A social network from a user history*: This social network shows the relationship among modellers who use the recommendation system. The focus of this second social network is different from the first one. In this social network the decision of users is the main driving factor for the generation of the social network. This social network is generated based on the user's history, which consists of the users and the names of their selected recommendations. From the recommendation history we can obtain three views on the social network. (i) According to the modelling purpose which may be documentation, analysis or execution. Before saving the process in the repository the user can annotate the process with this kind of information. If the user has annotated the process with the modelling purpose then social networks can be generated based on the recommendation history and focusing on the modelling purpose. (ii) According to process names which have been selected in an organisation (iii) Regarding preceding and subsequent process models. The recommendation system stores (for each modelling purpose e.g. analysis) the order of selected process model parts for each user. Based on this information a social network which considers the order of selected process model parts can be generated.

(3) *A social network from an insertion history*: This social network shows the relationship among modellers who decided for equal recommendations. As already mentioned before the recommendation system stores the order of inserted process model parts into the workspace of a user. In our scenario, a user can generate a new process model and insert it into the repository or s/he can search for appropriate process models and generate a new model by combining recommendations. From this information, a social network reflecting the insertion history of users can be generated. Figure 2 shows a social network, which has been generated from an insertion history.

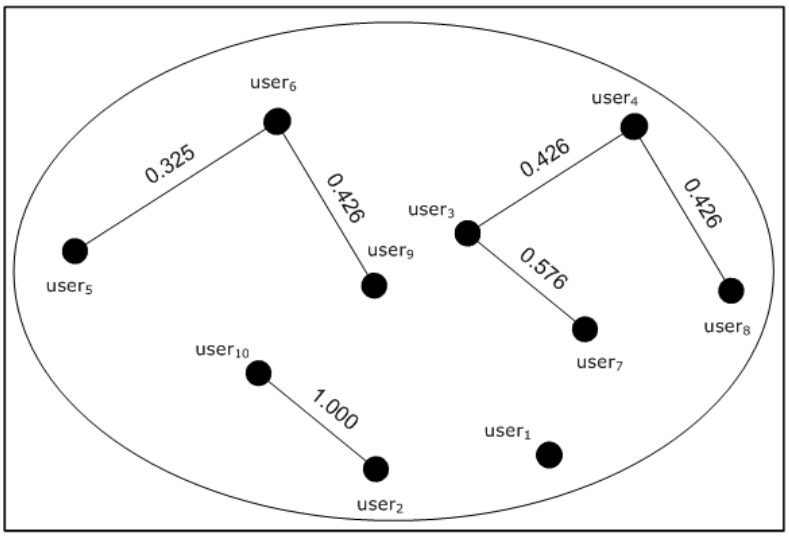

Figure 2: ***Example of a social network generated from insertion history***

In summary, these three social networks that are derived from existing process models stored in a repository and an insertion history may be used for the representation of the following new business knowledge. Firstly, the system gathers persons sharing same interests or supports strategic collaboration. Secondly, the "socially" extended recommendation-based modelling support system encourages user trust behaviour and participation by those users who are unskilled, because users gain insight who already selected specific processes. Thirdly, the system propagates process changes to users involved in the network.

Generally, the three types of social networks are subject to constant dynamic modifications, for example because a user will select new process models in her job. Conversely the design of business processes can be influenced by a social network. In Figure 2 the three process builders' user5, user6 and user9 are members of a clique. Additionally, user1 is new in the network and trusts user5. Consequently, user1 selects always the same process model part as user5 when creating e.g., approval processes. Thus, user1 will be once connected to the network of user5.

To inform user1 as soon user5 has e.g. updated process models, the recommendation system offers a push and a pull service. Push services in the recommendation system involve actively sending (or pushing) information to a specific process builder that the process builder knows to be interested in this information. The user can either push the information to her social networks, to process builders belonging to a clique or even to specific process builders. A pull service involves users that specified that they want to receive information if a certain process model has been changed. Consequently, the social network influences the design of business processes and the selection of recommendations (respectively business process models) influences the structure of social networks. Process mining techniques are used in this scenario to extract several groups in which people have a similar behaviour pattern or similar preferences. Based on this information customised push services can be provided.

4 Business process deployment/performance and Social Software

The paradigm of Business Process Management stresses the importance of integrating entire processes rather than simply integrating data or applications [18] [19]. The aim is to design and control the organisational structures in a flexible way so they can rapidly adapt to changing environments. During the early 90's, Workflow Management Systems (WFMS) have been developed as appropriate technological solutions for integrating process islands in order to collaboratively provide business solutions that each individual application is unable to do. However, the formalisms developed for the specification of workflow definitions were almost systematically activity oriented leading to process definitions which are

easily transformable in executable code but, in the same time, are prescriptive and rigid. This kind of specifications (activity driven) and the corresponding execution mechanisms (scheduler based) are convenient for well defined process models but not for knowledge intensive ones [20].

Flexibility has been the focus of many researches [20] [21] [22] [23]. It is defined in [24] as “the ability to yield to change without disappearing”. Business process flexibility means fast reactivity to internal and external changes. It also reflects the ability that the support systems have to take into account business changes. The necessary amount of flexibility depends on the *nature* of the business processes. Two categories can thus be differentiated. The first concerns well-defined and -often- repetitive processes having important coordination and automation needs. The second concerns ill-defined (often knowledge intensive) processes. The essential preoccupation with the latter is the information and knowledge sharing between the actors implied in the processes more than the coordination of their tasks. Business processes of this category require evidently more flexibility. For many organisations, well-defined and ill-defined processes coexist and should be handled in the final enterprise model [20] [25].

Several classifications have been proposed for workflow applications. The commonly used was defined in [26] and divides workflows into four classes, depending on the nature of the processes they support and the value these processes have for the enterprise:

- *Production workflows* involve repetitive and predictable BPs and implemnts the core processes of the enterprise. This is the closest category to the generic workflow product structure adopted by WfMC [27] .
- *Administrative workflows* involve repetitive, predictable processes with simple task coordination rules and do not concern the core processes of the enterprise.
- *Collaborative workflows* include iterative tasks over the same step until some form of agreement has been made. It seems very difficult to model those using classical WFMSs and the underlying activity-oriented (prescriptive) models since it is impossible to predefine the steps to follow. Most of the co-ordination is done by human participants.
- *Ad hoc workflows* have no predefined structure. Execution support is limited to the provision of communication mechanisms to route case (process instance) data between workers and possibly some support for logging and state tracking. Ad hoc workflows tend to be created to deal with exceptions. The coordination is controlled by human participants.

4.1 Social workflow systems

Considering flexibility and adaptability as the most stressed challenges in workflow management research, social software might contribute in several ways. First we can learn from social software in the sense that openness, discussion, broad participation rather than -or let us say- in addition to exclusiveness, instruction and expertise is an valuable source of knowledge and learning thus supporting management, and second we can transfer social technology to BPM software.

Looking from a high abstraction level, a wiki system and a WFMS represent opposite ends of a spectrum when considering their way of dealing with objects of work. While a wiki system has a web page (or a set of pages) as its typical target of work, a WFMS has to deal with a variety of business objects including orders, invoices, delivery notes, payments, goods receipts. While a wiki is typically accessible to the public, a WFMS is typically confined within organisational boundaries. While a wiki allows typically editing content pages equally to any person, WFMSs usually underlie strict policies determining who may change a workflow definition and who may access which business. While wikis invite anybody to participate in composing, reading and reviewing a web page, WFMSs assign people explicitly tasks according to their competencies and roles. While wikis are easy to use even for non technical oriented persons, WFMSs generally require deeper understanding of the business process models and technical constraints.

Finally, and that is in our opinion the most inspiring difference, wikis use the common wisdom of a community to reach completeness rather than expert knowledge to create sound workflow definitions.

The first wiki systems were focused on easy-to-use, quick collaborative editing of knowledge (e.g. Ward Cunningham and his WikiWikiWeb[2]) in the form of web pages on a dedicated website. Soon features for searching, revision management, enhanced formatting and linking appeared. With the evolution and growth of Wikipedia as the most prominent representative several new requirements occurred. Especially the huge amounts of articles created and the degree of participation in authoring and reviewing articles required more attention to version management and control. Although public participation in collecting knowledge remains a basic idea behind a wiki, business software industry put wikis in organisational context and developed wikis into a tool for team collaboration. Changing the target domain from public to an organisational context implies a change in access policies as well. Access policies have to comprise all operations on wiki content like in more traditional content management systems and must be modelled per wiki instance to fit both intra-organisational and inter-organisational demands of an application.

As for this quick analysis we can summarise that three dimensions may be considered when evaluating wiki-scenarios in organisations:

1. The degree of organisation of the involved community
2. The degree of specificity of wiki objects [28].
3. The degree of desired completeness

The first dimension reaches from bottom-up to top-down development of a shared knowledge space within an organisation or in the public and defines to which degree collaboration and access policies have to be introduced. The second dimension symbolises the data structure of a wiki-object and to what degree it underlies a formal definition.

As a third dimension, one could consider the degree of desired completeness or continuous evolution versus development of a final version (infinite vs. finite number of review cycles).

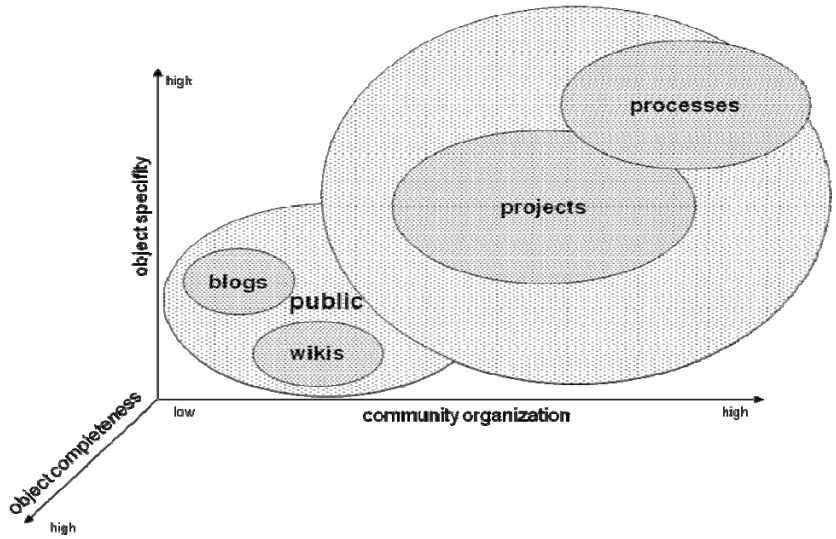

Figure 3: ***Dimensions defining wiki scenarios***

Figure 3 shows the three dimensions described and a possible value range. While wikis for web page creating and editing require not too much concern about consistency and have a simple underlying workflow, a sales order wiki would involve various assignees from different organisational units and would have to be based on strict transactional rules to ensure data consistency. In between these two extremes we

[2] Cunningham, W.: Wiki Wiki Web http://c2.com/cgi/wiki

might find numerous use cases where a moderate degree of completeness is required, a middle sised department is involved and collaborative development of a business document with a semi-formal structure (e.g. contract, project plan, business blueprint ...) is in focus exceptions and inaccuracy [29]. The web-based architecture of wiki-software is an ideal facilitator for exposing artefacts to a broad audience increasing the number of potential contributors.

While public wikis and team wikis for web based knowledge collections have become widely popular, using wiki-systems in business application areas is still uncommon.

With XoWiki [30] ContentFlow [31] which is based on the well known OpenACS [32] Community framework, an approach is presented how to link social capabilities of wikis with workflow management. In this implementation a wiki is used for both workflow design and enactment. Features like browser based modelling of a workflow, named objects to reference to objects (associated application interfaces, forms or pages), revision management, notifications, tagging and a graphical representation to evaluate the frequency of collaborative activities are implemented. Instances of a workflow and their states are stored persistently and may be used for reprocessing an application's former state or offer a base for recommendation-based workflow mining techniques [33] [34].

XoWiki ContentFlow shows how typical features of social software (wiki, community framework) are utilised in workflow management.

Considering again the need for flexibility and adaptability a wiki enabled workflow system seems to be the ideal framework to address problems of fast changing workflows. A workflow definition is a wiki object with a very low degree of completeness and requires high flexibility from its underlying workflow system. In highly dynamic workflow scenarios an a priori well modelled workflow might be obsolete. Exposed to a community via a wiki-based framework a high responsiveness to workflow changes will be reached and exceptions can be detected and repaired in a collaborative manner.

4.2 Workflow Enactment in a Social Software Environment

Using social software as an IT support mechanism for business processes is not a novel concept. Enterprise 2.0, for example, builds on using Web 2.0 social tools in a business context. In most of the existing instances, however, social software is used as a pervasive information sharing framework, but no advanced support for the enactment of the business process is made available.

Consider the following example: an organisation wants to manage the process associated to a photo contest using a forum. A possible solution can be as follows: when a new contest starts, a new thread is created in the contests forum. Participants can submit their photos by adding posts in this thread. The thread remains accessible for a specified amount of time (e.g. a week), after that the thread is locked and a new poll thread is created. This latter thread is used to collect the votes from other participants. After another specified amount of time, the poll thread is also locked and the poster that has received most of the votes is the winner. In this example social software tools have been used to share, in a structured container, the artefacts related to the contest (the photos, the votes, etc...). What's missing is the ability to support the users in answering the following questions: What has to be done? Who is in charge of doing it? When should it be performed?

WFMSs have been designed to provide IT support in answering these questions. But they have also been designed to operate in a completely different context. Most workflow management systems are prescriptive systems. Users are not only supported, they are enforced in performing tasks in specified sequences. Managing exceptions and adapting to changes is a well-known limiting factor for these systems. A social software environment is the place of collective intelligence, IT should support the knowledge workers not enforce their behaviour. We suggest an environment to support enactment of processes fostering the sharing of knowledge about the business best practices.

The first obvious mismatch between WfMS and these new tools is that a WfMS enacts a process on the basis of a well-defined process model. The set of actions that can/have to be performed by the actors change when the state of the process changes. In this context the state of the process is an abstraction addressed by a model (like a state of a finite state machine or a set of tokens in a Petri net).

In a social software context there is not a well-defined process model. The process can be changed at any given moment by the participants on the basis of their experience and of their knowledge. For this reason the term "organisational best practice" seems better appropriate than workflow in this context. Not having a predefined process model, however, does not mean the chaos, what usually happens is that, after the first few iterations, the process "takes shape", becoming more and more structured. Still it is possible to support the continuous evolution of a process and exception handling is much simpler. A point that is worth making is that missing a process model does not mean missing a process state. A state concept is needed if one wants to be able to help users in replying to the 'who', what, when questions. This if a process model is missing, the state as an abstraction is also missed but not the state of the process as a factual entity.

BPMN (Business Process Modelling Notation) [35] is a standardised graphical process notation that is experiencing a rapid adoption among BPM tools vendors. There is no need; however, to have a BPMN model of a conference review process to know that paper selection can start after all the reviews have been received. This is because the status of a process is inherent to the information associated to artefacts that are part of the process. The suggested approach is then to start from a representation of the state of the process as a collection of data and metadata (i.e., content and context). By extracting and aggregating all process-relevant information from the artefacts, we are able to define the current state of the process. The obvious next step is to take advantage of this knowledge to support users in applying the organisational best practices by suggesting them what should be performed at this point of the process and by supporting the automation of some of the interactions. Suggestions can be provided in the context of a process-aware recommender system, improving the adoption of the organisational best practices.

Automation can be supported in a peer-to-peer fashion; for example in Social X-Folders [36] an approach in this sense is proposed: since a process is enacted as a sequence of interactions among the actors, automation support is provided as a way for each actor to execute part of the tasks she/he is in charge of accomplish when the process reaches a given state.

In Social X-Folders aggregation of feeds (coming from various sources: forums, blogs, wikis, shared agendas, etc...) are used to expose process-relevant information and a reaction engine uses this knowledge to fire automated tasks on behalf of the users. These tasks mimic users' interactions with the (web-based) social software tools by playing-back the very same sequence of HTTP transactions taking place between the application server and the web browser. These sequences represent part of the process knowledge and they too can be shared by using social software tools [37].

This method, for example, has been applied to support the aforementioned photo contest process: the reaction engine monitors a calendar feed and a feed reporting the status of the contest in order to execute the creation of new voting threads and the locking of submission threads when due.

5 Dimensional view of utilising social software within enterprises

In today's fast changing business, companies need to communicate directly with partner and customers and adapt results quickly in the daily business. Social software tools and web application support this and their usage in business processes will change these processes themselves. Figure 4 gives an overview about fields of change in an enterprise. All organisational departments are influenced, internal as well as external operating units; management processes as well as production and support processes.

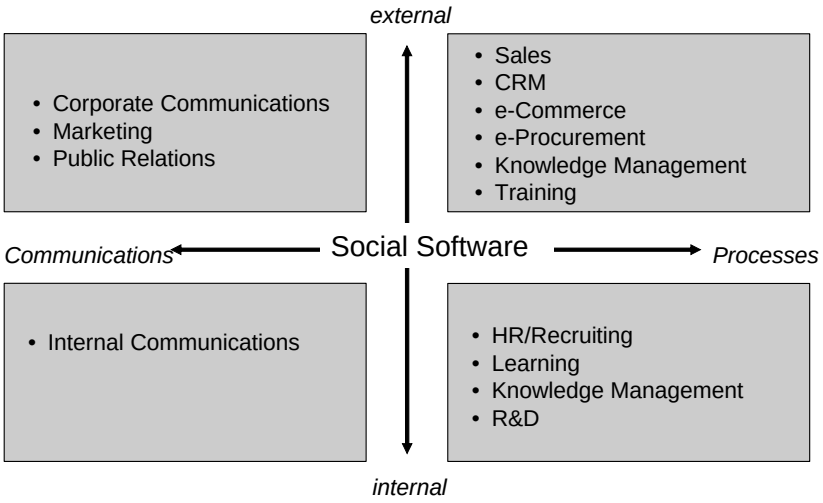

Figure 4: Social software influence in an enterprise [38]

Main effects of social software will be recognised in all knowledge related processes. This can be sales, marketing, innovation or even human resources or other. Relevant is the question, which are the core processes and where is most potential to improve communication and cross linking. So this can be HR (if it is a HR supporter) or it is sales (as in a company in a market with many similar providers) or (which is true for most companies) innovation? [39] [40]. Following some examples are described.

Some enterprises have gained a competitive advantage by having a community of innovative users connected with the enterprise's product [40] [41]. Such user involvement may add value in several ways. For instance, social software has been found to enable the end users to provide peer support [42] and innovations [43] regarding the enterprise's product. For some enterprises, social software has proved to be useful in their brand building. Such utilisations of social software have enabled the enterprises to sense market forces with unprecedented accuracy and efficiency and allowed them to respond to nuances in conversations that hint at unarticulated needs [44]. It has also been acknowledged that social software enables international reach - that is, help companies gain access to potential customers and co-developers all over the world [45]. Furthermore, some enterprises have begun to explore the idea of utilising social software to guide their product development [46]. The common theme appears to be that social software may turn into a strategic asset: "an imperfectly imitable resource that can hardly be purchased but must evolve" [41].

Despite the potential that social software has already demonstrated, enterprises are still struggling with the challenge of how to benefit from social software in practice. Enterprises have found it difficult to utilise social software in such manner that it 1) achieves its objectives, 2) adds value and is attractive to the members and 3) avoids unintended consequences. Furthermore, since the utilisation of social software often initiates a radical transformation of customer-producer relationships [39], the enterprises are likely to be forced to reconsider their business processes.

These challenges have led us to propose a dimensional view illustrating different utilisations of social software and their relationship to the enterprise's business processes (Figure 5) [47]. The first dimension, 'targeted stakeholders', describes to whom the particular instance of social software is intended for. We have discovered that the targeted stakeholders may range from enterprise's internal stakeholders to the end-users of a product. The other dimension, 'nature of objectives', describes the purpose of the social software solution. We have identified differing objectives for social software ranging from the support of product development related collaboration to supporting business-related aspects of the enterprise's product. Based on these two dimensions, we have formed four categories and labelled them with business processes they are likely to affect: 1) brand building, 2) distributed product development, 3) sales & partner support and 4) user involvement.

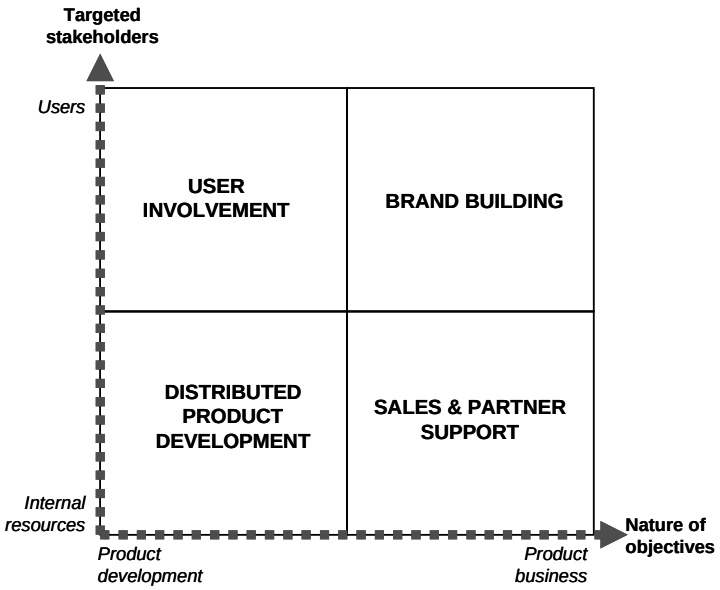

Figure 5: ***Dimensional view illustrating different utilisations of social software and their relationships to business processes***

We believe that the proposed dimensional view is valuable because it provides an overview of how enterprises have benefited from utilising social software. To be more precise, the dimensional view helps to determine the business processes that are likely to benefit from utilising social software and gives some guidelines on what kind of social software solution would be suitable to support a particular business process. Since the intended purpose and targeted stakeholders appears to have an effect on how the sociability of the intended solution should be supported, we hope that the dimensional view will be further developed to become a useful tool guiding the enterprises to the successful path in their own utilisation of social software.

6 Considerations about Digital Identity and Reputation in business processes using social software

As has been mentioned in the previous sections of this paper, there are many motivating factors for the integration of social software as part of business practices. In the enterprise context, the appeal of integrating social concepts is twofold: there is the opportunity to improve business processes through richer, socially enabled software interactions and to create mechanisms for the human agents within the enterprise to add value to the knowledge of the company [48]. The Enterprise 2.0 term [5] [49] is gaining adoption.

When integrating social software into business processes, there is the potential for a fundamentally deeper understanding of the individual within the enterprise. The overarching concern of understanding the people behind the data is the same whether social software is being applied in the Service Oriented Architecture (SOA) space or in a discrete wiki or blog. This fundamental issue comes under the name digital identity. In the context of this paper, the term digital identity is used to express the concept of a cross-set unique token. In any system, or set of integrated systems, used within a business process, establishing the exact identity of a human agent within that system, or subset thereof, is critical.

When using social software data artefacts, having specific knowledge of who created that data, via a unique token or digital identity, allows a process to link a human agent to a specific body of work or expertise. If the enterprise were a green field environment, an integrated identity solution could be used. The more pragmatic approach in a real world scenario will be that of many legacy data artefacts (mailing lists, commit logs) with the addition of new, best of breed social applications. For a unified view of the identity

of human agents within such an enterprise, new techniques must be applied to analyse the existing data sets, and the data for potential pseudonyms and present a digital identity resource from which to make human agent based assertions. In section 2.2, a recommender in the context of social software was presented. This process of looking for connections between users of the system is dependent on a solid foundation of Digital Identity from which to make assertions.

Recommenders can also act as a user-based filtering mechanism. As the adoption of social software within the enterprise domain becomes more prevalent, so comes the issue of too much information. As more users generate more blog posts, wiki edits and messages to mailing lists, this can lead to information overload [50]. Studies have been performed that show if people are subjected to an overly dense information stream [51], they will be less productive, and in the worst case, abandon the system altogether. Any system hoping to provide information to the user needs to consider this issue. Rather than modelling data connections based on existing physical systems, computer based mechanisms can act as a filtration system, distinguishing good information from less important.

Social software needs to provide contextually useful information connecting specific users to each other in order to respond to environmental information and subsequent business process exceptions. During the process of integrating social software (and existing social data artefacts such as mailing lists) within the enterprise, this overload issue may be mitigated by unifying identities within disparate silos of information in order to ascertain relationships between individuals and data. From this unified identity, less relevant data can be occluded from certain users, effectively bubbling up more relevant content and agents.

As was mentioned in the opening section of this paper, in the context of social software, trust is a complex issue as users of the social software may not know each other. For an enterprise to integrate social software into business processes there is the need for both reputation, trust and an authoritative voice [52]. Without these tenets, there can be little value added to the enterprise.

Trust and reputation are subjective measures, as both are based entirely upon personal feelings and the interpretation of ambiguous signals [53], rather than the objective representation of fact. These facets are made more difficult as, in the context of social software; it is highly likely that the users of the system will never meet. Trust may be viewed as a function of the agent's desire for an outcome in relation to their perception of the transactional risk dependent upon that agent's attitude towards risk in a specific context [54]. This measure may alleviate concerns of opportunistic behaviour [55] from the other participant in a given transaction. This opinion led abstraction of a deficit of information [56] can form the foundation of a decision making process. The act of aggregating an individual user's interpretation of ambiguous or asymmetric knowledge [57] can lead to a broader context from which to make a decision.

Without a clear sense of identity, there can be no foundation for trust or reputation. In the enterprise environment, trust and reputation will also become a matter of key concern with the adoption of social software. Without a clear concept of identity across data artefact sets, there can be no concept of a unified user reputation. In order for enterprise to leverage the social graph to integrate users in business processes in a more meaningful manner, thought needs to be given for a mechanism to create a unified digital identity resource in an automated manner. From this resource, foundations for trust and reputation can be built and this will enable business processes and social software to have a richer source of information from which to make assertions about users of a system.

7 Conclusion

Combining business process management and social software offers a number of benefits. Social software allows integrating users into business process management. The threshold to provide information and knowledge to the design, implementation and optimisation of processes is lowered or even abandoned. The basic principles openness and ease of usage are the pillars of the wide acceptance of social software seen mainly in the private sector. Therefore, prospecting similar effects in the business environment means

changing communication principles away from predefined, hierarchic communication structures. Furthermore, the divide between abstract process models, lifecycles, evaluations and the executed processes, can be narrowed or even completely avoided. The lack of formal barriers also tears down psychological barriers. Resistance is supposed to be lower due to a low entrance barrier. Instead, due to the immediate effects of employee action, their involvement and commitment may be increased. Therefore, social software has the potential to enhance collaborative and knowledge intensive business processes by improving the exchange of knowledge and information, to speed up decisions, to improve the global reactivity of the enterprise.

Combining business process management and social software offers new opportunities for the design of business processes. Thus, when designing a business process one should consider how it can be supported with the novel instruments provided by social software. For this purpose, a number of methodological guidelines have been provided.

To exploit advantages of social software services in BPM, a recommendation-based modelling support system has been enhanced with social network features. The core of the recommendation system takes into account a process builder's modelling context and the modelling history of a community of users, which suggests process model parts to the user that may help him achieving an individual modelling goal.

Wiki enabled workflow system seems to be the ideal framework to address problems of fast changing workflows. In highly dynamic workflow scenarios an a priori well modelled workflow might be obsolete. Exposed to a community via a wiki-based framework a high responsiveness to workflow changes will be reached and exceptions can be detected and repaired in a collaborative manner.

A dimensional view is valuable to clarify the benefits from combining social software and business process management. To be more precise, the dimensional view helps to determine the business processes that are likely to benefit from utilising social software and gives some guidelines on what kind of social software solution would be suitable to support a particular business process.

The benefits of combining business process management and social software are facilitated by the completely new approach for putting together the inputs of different people. Instead of predefining the inputs of all participants in a top-down manner, all stakeholders are encouraged to provide their inputs without the existence of an overall plan in a bottum-up manner. Content creators are not predefined, each user can add context – by tagging, evaluating, commenting or even reading. The sum of all these interactions is a new content itself and part of the collective intelligence. But if everybody can – and should - annotate and include new pieces of content: how is quality and trust ensured? E.g., if the finance process is commented and suggestions for improvement are made by each user, how is legal accordance assured? Building difficult checking processes can’t be the answer as effects of speed, feedback, authenticity and directness are ignored and so one motivation of active usage is destroyed. New kinds of risk management and governance rules are needed with different levels of inference and strictness.

8 References

[1] R. Schmidt and S. Nurcan, “BPM and Social Software,” *BPM2008 Workshop Proceedings, Springer – LNCS*, Mailand: 2008.

[2] V. Bush, “As we may think,” *interactions*, vol. 3, 1996, pp. 35-46; http://portal.acm.org/citation.cfm?id=227186.

[3] M. Granovetter, “The Strength of Weak Ties,” *The American Journal of Sociology*, vol. 78, 1973, pp. 1360-1380; http://dx.doi.org/10.2307/2776392.

[4] M. Granovetter, “The strength of weak ties: A network theory revisited,” *Sociological Theory*, vol. 1, 1983, pp. 201-233.

[5] A.P. Mcafee, “Enterprise 2.0: the dawn of emergent collaboration,” *Engineering Management Review, IEEE*, vol. 34, 2006, pp. 38-38.

[6] G. Hamel and B. Breen, *The Future of Management*, Harvard Business School Press, 2007.

[7] Y. Benkler, *The Wealth of Networks : How Social Production Transforms Markets and Freedom*, {Yale University Press}, 2006; http://www.amazon.ca/exec/obidos/redirect?tag=citeulike09-20&path=ASIN/0300110561.

[8] U. Cress and F.W. Hesse, “Knowledge sharing in groups: experimental findings of how to overcome a social dilemma,” *Proceedings of the 6th international conference on Learning sciences*, International Society of the Learning Sciences, 2004, pp. 150-157.

[9] A. Stocker, M. Strohmaier, and K. Tochtermann, “Studying Knowledge transfer with Weblogs in Small and Medium Enterprises: An Exploratory Case Study,” *Scalable Computing: Practice and Experience* , vol. 9, 2008.

[10] M. Granitzer, G. Granitzer, K. Tochtermann, S. Lindstaedt, A. Rath, and W. Groiß, “Automating Knowledge Transfer and Creation in Knowledge Intensive Business Processes,” *Proceedings of the First Workshop on Business Process Management and Social Software BPMS2'08 in conjunction with 6th International Conference on Business Process Management*, 2008.

[11] N. Oliver, G. Smith, C. Thakkar, and A.C. Surendran, “SWISH: semantic analysis of window titles and switching history,” *Proceedings of the 11th international conference on Intelligent user interfaces*, ACM New York, NY, USA, 2006, pp. 194-201.

[12] M. Granitzer, M. Krull, C. Seifert, A. Rath, N. Weber, O. Dietzel, and S. Lindstaedt, “ Analysis of Machine Learning Techniques for Context Extraction,” *Proceedings of 2008 IEEE International Conference on Digital Information Management (ICDIM08)*, 2008.

[13] B.J. Rhodes and P. Maes, “Just-in-time information retrieval agents,” *IBM Systems Journal*, vol. 39, 2000, pp. 685-704.

[14] W.M.P. van der Aalst, H.A. Reijers, A. Weijters, B.F. van Dongen, A.K. Alves de Medeiros, M. Song, and H.M.W. Verbeek, “Business process mining: An industrial application,” *Information Systems*, vol. 32, 2007, pp. 713-732.

[15] T. Hornung, A. Koschmider, and G. Lausen, “Recommendation Based Process Modeling Support: Method and User Experience,” *Conceptual Modeling-Er 2008: 27th International Conference on Conceptual Modeling, Barcelona, Spain, October 20-24, 2008, Proceedings*, Springer, 2008, p. 265.

[16] A. Koschmider, M. Song, and H.A. Reijers, "Social Software for Modeling Business Processes," *Proceedings of the First Workshop on Business Process Management and Social Software BPMS2'08 in conjunction with 6th International Conference on Business Process Management*, 2008.

[17] M. Song, "Organizational mining in Business Process Management," 2006.

[18] R.T. Burlton, *Business Process Management: Profiting from Process*, Sams, 2001.

[19] W.M.P. van der Aalst, J. Desel, and A. Oberweis, *Business Process Management, Models, Techniques, and Empirical Studies*, Springer-Verlag London, UK, 2000.

[20] S. Nurcan, "A Survey on the Flexibility Requirements Related to Business Processes and Modeling Artifacts," *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, 2008, p. 378.

[21] O. Saidani and S. Nurcan, "Towards Context Aware Business Process Modelling," *The 8th Workshop on Business Process Modelling, Development, and Support (BPMDS'07,(in association with CAISE'07), Springer Verlag (pub), June 11-12, 2007, Trondheim, Norway*, 2007.

[22] R. Schmidt, "Flexible Support of Inter-Organizational Business Processes Using Web Services," *Proceedings of the CAiSE'05 Workshops*, Porto: 2005, pp. 51-58.

[23] D. Shi and R.L. Daniels, "A survey of manufacturing flexibility: Implications for e-business flexibility," *IBM SYSTEMS JOURNAL*, vol. 42, 2004, pp. 414-427.

[24] G. Regev and A. Wegmann, "A Regulation-Based View on Business Process and Supporting System Flexibility," *Proceedings of the CAiSE*, 2005, pp. 91-98.

[25] S. Nurcan, "Analysis and design of co-operative work processes: a framework," *Information and Software Technology*, vol. 40, 1998, pp. 143-156.

[26] G. Alonso, D. Agrawal, A. El Abbadi, and C. Mohan, "Functionality and Limitations of Current Workflow Management Systems," *IEEE Expert*, vol. 12, 1997, pp. 105-111.

[27] Workflow-Management-Coalition, "Terminology & Glossary," *Document Number WFMC-TC-1011, Feb*, 1999.

[28] A. Bernstein, "How can cooperative work tools support dynamic group process? bridging the specificity frontier," *Proceedings of the 2000 ACM conference on Computer supported cooperative work*, ACM New York, NY, USA, 2000, pp. 279-288.

[29] F.B. Viégas, M. Wattenberg, and K. Dave, "Studying cooperation and conflict between authors with history flow visualizations," *Proceedings of the SIGCHI conference on Human factors in computing systems*, ACM Press New York, NY, USA, 2004, pp. 575-582.

[30] G. Neumann, “XoWiki - Towards a Generic Tool for Web 2.0 Applications and So-cial Software,” 2007.

[31] G. Neumann and S. Erol, “From a social wiki to a social workflow system,” *Proceedings of the First Workshop on Business Process Management and Social Software BPMS2'08 in conjunction with 6th International Conference on Business Process Management*, 2008.

[32] “OpenACS Home”; http://openacs.org/.

[33] W.M.P. van der Aalst, V. Rubin, B.F. van Dongen, E. Kindler, and C.W. Gunther, “Process Mining: A Two-Step Approach using Transition Systems and Regions,” *BPM Center Report BPM-06-30, BPMcenter. org*, 2006.

[34] B. Weber, B.F. van Dongen, M. Pesic, C.W. Gunther, and W.M.P. van der Aalst, *Supporting Flexible Processes Through Recommendations Based on History*, Beta, Research School for Operations Management and Logistics, 2007.

[35] “BPMN Information Home”; http://www.bpmn.org/.

[36] D. Rossi and F. Vitali, “Workflow Enactment in a Social Software Environment,” *Proc. BPMS2 Workshop, Springer – LNCS, to appear*.

[37] G. Leshed, E.M. Haber, T. Matthews, and T. Lau, “CoScripter: automating & sharing how-to knowledge in the enterprise,” 2008.

[38] S. Happ, F. Schönfeld, and U. Volejnik, “Vom Web 2.0 zum Unternehmen 2.0: Kommunikation und Prozesse im Wandel,” *Konferenzband KnowTech 2007*.

[39] S. Nambisan, “Designing virtual customer environments for new product development: Toward a theory,” *The Academy of Management review*, vol. 27, 2002, pp. 392-413.

[40] E. Hippel, “User toolkits for innovation,” *An International Publication of the Product Development & Management & Association*, vol. 18, 2001, pp. 247-257.

[41] L.B. Jeppesen and L. Frederiksen, “Why Do Users Contribute to Firm-Hosted User Communities?,” *Organization Science*, vol. 17, 2006, pp. 45-63.

[42] L.B. Jeppesen, “User Toolkits for Innovation: Consumers Support Each Other,” *Journal of Product Innovation Management*, vol. 22, 2005, pp. 347-362.

[43] E. von Hippel, *Democratizing Innovation*, MIT Press, 2005.

[44] G. McWilliam, “Building Stronger Brands through Online Communities,” *SLOAN MANAGEMENT REVIEW*, vol. 41, 2000, pp. 43-54.

[45] J. Preece, *Online Communities: Designing Usability, Supporting Sociability*, John Wiley, 2000.

[46] S. Lewis, “Using online communities to drive commercial product development,” *In Proceedings of the Human Factors in Computing Systems archive (CHI '08)*, 2008.

[47] S. Jantunen, “Utilizing Firm-hosted Online Communities in Software Product Business: A Dimensional View,” *Proc. BPMS2 Workshop, Springer – LNCS, to appear*.

[48] B. Jennings and A. Finkelstein, “Digital Identity and Reputation in the Context of a Bounded Social Ecosystem,” *BPM2008 Workshop Proceedings, Springer – LNCS, Milano, 2008.*; https://pandora.informatik.htw-aalen.de/twiki/pub/BPMS2/WebHome/3_Jennings.pdf.

[49] Y. Kakizawa, “In-house use of web 2.0: Enterprise 2.0,” *NEC TECHNICAL JOURNAL, Jan 2007*.

[50] S.R. Hiltz and M. Turoff, “Structuring computer-mediated communication systems to avoid information overload,” *Communications of the ACM*, vol. 28, 1985, pp. 680-689.

[51] A. Mehrabian, “A questionnaire measure of individual differences in stimulus screening and associated differences in arousability,” *Journal of Nonverbal Behavior*, vol. 1, 1977, pp. 89-103.

[52] R.S. Burt, “A note on social capital and network content,” *Social Networks*, vol. 19, 1997, pp. 355-373.

[53] C. Fombrun and M. Shanley, “What's in a name? Reputation building and corporate strategy,” *Academy of Management Journal*, vol. 33, 1990, pp. 233-258.

[54] L.F. Luna-Reyes, A.M. Cresswell, and G.P. Richardson, “Knowledge and the development of interpersonal trust: a dynamic model,” *System Sciences, 2004. Proceedings of the 37th Annual Hawaii International Conference on*, 2004, pp. 86-97.

[55] J.L. Bradach and R.G. Eccles, “Markets versus hierarchies: from ideal types to plural forms,” *Annual Review of Sociology*, vol. 15, 1989, pp. 97-118.

[56] G. Möllering, “The Nature of Trust: From Georg Simmel to a Theory of Expectation, Interpretation and Suspension,” *Sociology*, vol. 35, 2001, pp. 403-420.

[57] G. Akerlof, “ The market for lemons ; quality uncertainty and the market mechanism,” *Quarterly Journal of Economics*, 1984, pp. 488-500.