Combining BPM and Social Software: Contradiction or Chance?

Selim Erol, Vienna University of Economics and Business, 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

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

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'Informazione Università di Bologna – Italy

Rainer Schmidt, HTW-Aalen,
Rainer.Schmidt@htw-aalen.de
Anton-Huber-Str. 25
73430 Aalen, Germany

Keywords: Social Software, Business Process

Abstract

Social software received much attention in academia and industry due to many success stories. However, although social software is used widely 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 (BPMS2'08), as part of the International Conference on Business Process Management in Milano, show the manifold possibilities of joining concepts from Business Process Management and social software. Social software provides a better integration of all stakeholders into the business process life-cycle and offers new possibilities for a more effective and flexible design of business processes. The modelling of business processes may profit particularly from using social software techniques by alleviating the integration of process knowledge from all stakeholders. Also the implementation and deployment phase of the business process lifecycle may profit from social software by collecting valuable information for continuous process improvement from a larger set of sources than before. Furthermore, social software environments may be used to provide workflow support. Furthermore, 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, combining the input from independent contributors without predetermining the mechanism by which this is accomplished. An example of such social production may be seen in Wikipedia. Although the success of social software is rather new, its roots can be traced to 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 of implementing them using software systems.

The use of social software provides impressive results 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 do not necessarily know each other and are not organised a priori 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. This, however, presumes a high motivation of the participants [5], which is not always assured. Therefore, incentives may be necessary to ramp up the usage of social software. Furthermore, appropriate coordination mechanisms between the editors have to be chosen [6].

The content created in social software is under continuous assessment of all users. Every user may detect and correct flaws in the content, without using a formalised change procedure. This intense surveillance of changes is able to compensate for some issues created by the openness of social software, such as vandalism [7]. However, it is dependent on the balance of power between the attacker and the community whether false information can be detected and corrected.

Trust and reputation play a crucial role in social software. Changes are not initiated or authorised by hierarchic structures, but granted to (nearly) everybody, based on the assumption that nobody wants to damage their own reputation. Thus, all stakeholders can define content and content structures and self-organisation can be used more widely. However, due to malicious users and users overestimating their expertise, the reliability of information provided in social software remains an issue.

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 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 may 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* may be used as a component of a trust decision 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 [8]. It supports new communication patterns between customers and the enterprise. Multidirectional communication replaces the former unidirectional one from the enterprise to the customer. Using blogs, the customer may provide information to the enterprise, capturing and sharing 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.

1.1 Problem Description

At various conferences such as the International Conference on Business Process Management (BPM) or the Workshop series on Business Process Modelling, Development, and Support (BPMDS) there is a growing concern about the further development of Business Process Management (BPM) [9]. A number of phenomena and issues has been identified in the Workshop on Business Process Management and Social Software (BPMS2) [1]

Model-Reality Divide: The so-called model-reality divide 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 enactment of business processes: the modelled and the executed (real) processes fail to be adopted. Unsurprisingly, the employees do not accept such business process models but follow ("live") their own processes.

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

The roots of the model-reality-divide and lost innovation are manifold and interconnected as shown in Figure 1. They will be analysed in detail.

Information pass-on threshold: Ideas for improvement are not passed on to those responsible because this creates too much effort for the process owner and/or the user ("Why should I write a memo or a letter?"). Further processing is not transparent to them ("What will happen with my suggestion?") or success is considered to be improbable ("Will not succeed anyway"). A pass-on-threshold for information may also be created when the entering of information is strongly regulated, for example the process of change submission is too restrictive or simply takes too long due to approval steps. This means that users cannot submit their ideas easily. Therefore, important and valuable information is lost and potential improvements remain unrealised. As a consequence, the implemented process differs more and more from best practices and employees tend to execute their own "private" process which may contain optimisations which they have regarded as too difficult to integrate in the "official" process. Thus 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, especially the users, are properly involved with 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.

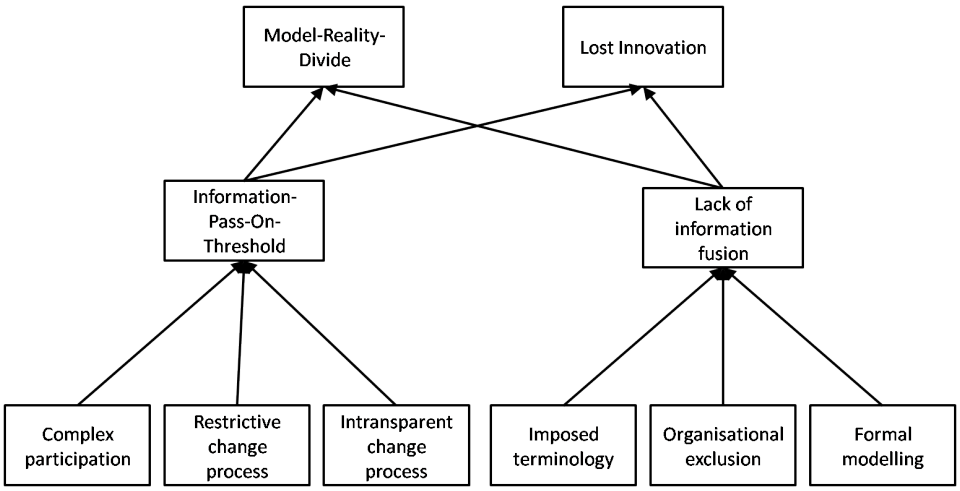

Figure 1: Roots of the model-reality-divide and lost innovation

1.2 Contributions

The methodological approaches behind social software present an ideal complementary solution space to the model-reality-divide and could capture lost innovation to business processes. 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. This process will also address another issue in the model-reality-divide, that of the lack of information fusion. Social software allows the group construction of a common terminology. Due to the egalitarian nature of such software, organisational exclusion may be avoided. Furthermore, the use of natural language avoids the creation of participation thresholds due to the use of formal modelling tools.

However, these fundamental benefits of social software have not been concretised for BPM. Therefore, combining concepts from Business Process Management (BPM) and social software will be the theme of this paper. The contributions of this paper are:

- Presentation of new opportunities provided by social software for the design of business processes
- Discussion on the benefits of social software in the business process life-cycle
- Discussion of digital identity and reputation in business processes using social software

The paper proceeds as follows: first, support which may be provided by social software to business process (re)design are presented. The next section is devoted to concepts of social software that may be applied to the deployment and enactment 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 New opportunities provided by social software for the design of business processes

In this section, approaches combining social software with BPM in order to support the business process design phase are presented. The following sub-section initially discusses the introduction of social software into Business Process Management System support.

2.1 Designing and managing business processes using social software

A process can be designed to take advantage not only 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 trained to perform one task 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 coordinate their work. Typically, the aim of this work structure 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 a logical order between tasks but this is not as rigid as at the assembly line. The construction of such business processes are goal driven and built around a desired delivery date for a specific artefact. 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 and services 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 may be formulated explicitly.

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

Both traditional BPMS and social software can address the management of work activities [10]. But social software provides a number of new tools:

- *Self identification*. Any actor who would like to contribute to an activity may do so and thus identifies themselves as competent to carry out such activity.
- *Transparency*. All work results are openly available to anyone.
- *Signing*. The performing actor signs all work activities upon completion.
- *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 may be banned.

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

- *Avoid the use of control flow*. Use the control flow mechanisms of BPMS primarily for controlling management activities and instead 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 who may contribute to the processes as well as histories of previous process executions.
- *Design 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 hand, participation is voluntary. A compromise is to make use of the notion of honour points, i.e. a participant receives credits in the form of honour points for activities they carry out and will be rewarded when they have obtained a certain amount of points. Such rewards may range from informal acknowledgements, monetary reimbursements or 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 beneficial when business processes concern production of non-standardised goods and services. There are also advantages 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 [11]. There is also a need for novel tools that provide both BPMS and social software mechanisms.

2.3 Automating Knowledge Transfer and Creation in Knowledge Intensive Business Processes

One critical factor in utilising technologies for organisational knowledge exchange is the well-known dilemma of knowledge sharing. Viewing this as a cost problem, employees try to minimize their cost for sharing knowledge with others while maximising 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 such knowledge may be transferred to a corporate setting [12], [13]. 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 of effort? For example, users usually have hard times explaining step-by-step how they compiled a report or what information sources they used for retrieving the required background information. However, 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 of sharing knowledge. With the aim of transferring the social software paradigm to business processes, one has to consider the means of reducing the cost of sharing successful and efficient process execution patterns as well as resources needed in the execution of such process patterns.

As outlined in [14], capturing the work context allows gathering patterns of weakly structured, knowledge intensive business processes executed in day-to-day businesses. At the heart of this aggregation system (see [14]) lies the context detector. Key strokes, mouse moves, resources touched and application switches a user performs in their daily work are gathered and aggregated by the context detector. Based on the aggregated information and previously learned models, machine learning techniques determine the current task in which a user is engaged. Experiments show a satisfactory accuracy of 75% [15], [16] 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 a list of tasks in which a user is engaged and resources relevant for such 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 substitutes for tags in folksonomies: a personal, user-generated description of resources. As opposed to tags, tasks are generated based on human trained models, thereby avoiding the problem in describing tasks with one single label. As has been seen in the process of comparing folksonomies to ontologies, the task-folksonomy can be seen as an informal representation of process knowledge compared to the formal representation normally used in business processes management. Capturing such work context, in an analogous manner to the output from a folksonomical approach, may lead to a broader understanding of the required tasks and resources from a sharing perspective.

- *Sharing of tasks and resources among users*: Users may search for tasks produced by 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 [17], tasks and resources suitable for the current work context of a user may be provided automatically. Also, knowing other employees with similar tasks may lead to ad-hoc groups and informal discussions on sharing workflow experiences outside the digital world.
- *Enrichment of resources*: Resources often used together in particular tasks allow the inference that they share something common, at least on a statistical level. 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 content and by creating relationships between resources based on the observed usage.
- *Collaborative creation of knowledge intensive ad-hoc processe*s: Task-folksonomies provide rich grounds for business process engineers to analyse tasks in which employees are engaged as well as possible dependencies amongst tasks. Having a substantive task repository increases the success of applying statistical process mining methods like those outlined in [18].

Although the creation of tasks-folksonomies seems to be a fruitful approach for bootstrapping the sharing of tasks and resources among employees and more generally in overcoming the knowledge sharing dilemma, critical aspects are still open. Privacy is an obvious concern. In [14]

a high user acceptance of the system could be achieved by allowing users 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 satisfactory 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.

2.4 Enhancing business process modelling using social software

To exploit the 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 [19] 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 achieve an individual modelling goal. For this the modelling support system works on top of a repository, which stores business process models (specifically process model parts) previously designed and stored by process builders 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 [20], process builders can gain insight on 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,; 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 [21] 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 between two activities executed by performer i. The cooperation metrics count 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 may 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) A second view may be from the perspective of process names which have been selected in an organisation.

(iii) The third view considers 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 that 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 on equal recommendations. As already mentioned earlier, 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 they 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.

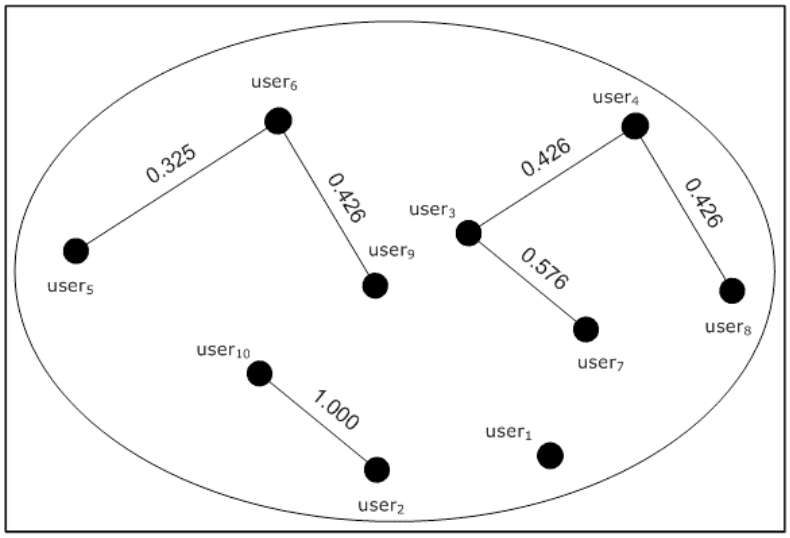

Figure 2: ***Example of a social network generated from insertion history***

In summary, these three social networks that are derived from existing process models stored in a repository and the process builder's insertion history may be used for the representation of the following new business knowledge. Firstly, the system gathers persons sharing same interests supporting 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 from experienced users engaged in such 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 their job. Conversely, a social network can influence the design of business processes. 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 always selects the same process model part as user5 when creating, for example, approval processes. Thus, user1 will be connected to the network of user5.

To inform user1 as soon user5 has, for example, updated process models, the recommendation system offers a push and a pull service [22]. 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 such information. The user can either push the information to their social networks, to process builders belonging to a clique or even to specific process builders. A pull service involves users who specified wanting to receive information if a certain process model

were to be changed. Consequently, the social network influences the design of business processes and the selection of recommendations (specifically 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.

2.5 Drawbacks and Pitfalls

Previous sections made clear that business process design may benefit in a variety of ways from the use of Social Software paradigms and Web 2.0 technologies. But are there any drawbacks?

Similar to the Web, several downsides exist. First and foremost, information/data quality tends to be lower while data quantity increases exponentially. According to numbers provided by Technorati [23], the blogosphere doubled every 5.5 month in 2006. Splogs, so called spam blogs, decrease information quality through spamming blog entries for advertisement or search engine optimisation. Hence, credibility and trust estimations for blogs are required. In using Web 2.0 technologies and Social Software paradigms for business process design, similar quality assurance factors have to be considered. Especially if one relies on automatic methods, a number of expert users may be required to assure quality.

Different from the Web, enterprises usually have a smaller contributing user base than web sites like Wikipedia. In Wikipedia, roughly 300 million page edits have been performed until 2008[24] with an average of 18.23 edits per page; an impressive number of corrections undertaken by a huge user base. However, the fact that only 1.7% of the registered Wikipedia users are actively participating showing potential for growth. Having only a small number of highly active users is a valid scenario for enterprises in utilising social software paradigms. Nevertheless, it will be an important task to develop an incentive system to persuade the employees to contribute. Social software provides its benefits only after some time. Therefore, the employees have to be persuaded to "invest" time and effort for some time to achieve benefits in the future.

Overall, by carefully paying attention to lessons learned from the Web, drawbacks maybe overcome. Motivating expert users for quality control and getting an active user base seem to be key ingredients to successful applications of Web 2.0 technologies and Social Software paradigms for business process modelling.

In the next section the support of social software for business process deployment and performance will be investigated. First, an approach is presented which links the social capabilities of wikis with workflow management. A wiki is used for both workflow design and enactment. Second, a new view on processes is used to deploy business processes. 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 they are in charge of accomplishing when the process reaches a given state.

3 Business process deployment/enactment and Social Software

The paradigm of BPM stresses the importance of integrating entire processes rather than simply integrating data or applications [25] [26]. 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 90s, Workflow Management Systems (WFMS) have been developed as appropriate technological

solutions for integrating process islands, in order to provide collaboratively business solutions which each application individually 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, at the same time, are prescriptive and rigid. These 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 [27].

Flexibility has been the focus of many researches [27] [28] [29] [30]. It is defined in [31] 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 which 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 evidently require more flexibility. For many organisations, well-defined and ill-defined processes coexist and should be handled in the final enterprise model [27] [32].

Several classifications have been proposed for workflow applications. One commonly used classification was defined in [33] 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 implements the core processes of the enterprise. This is the closest category to the generic workflow product structure adopted by WfMC [34] .
- *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.

3.1 Social workflow systems

Considering flexibility and adaptability as the most difficult aspects of 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 exclusiveness, instruction and expertise is a valuable source of knowledge and learning, thus supporting management, and secondly, 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, and goods receipts. While a wiki is typically accessible to the public, a WFMS is typically confined within organisational boundaries. While a wiki allows the editing of content pages equally by 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 explicitly assign people 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, the most interesting difference is that 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[1]) 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. The substantive amount 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 as 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, it can be summarised 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 [35].
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).

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

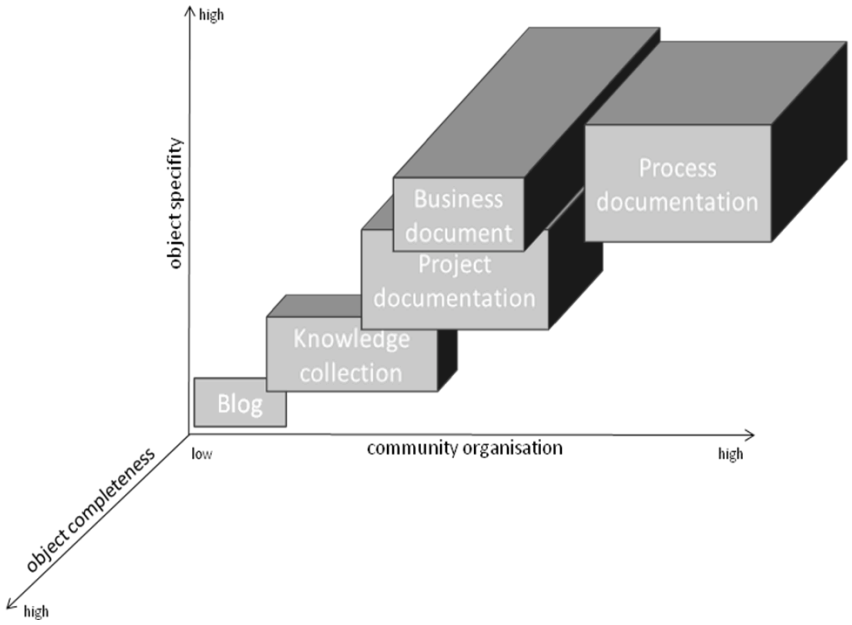

Figure 3: ***Dimensions defining wiki scenarios***

Figure 3 shows the three dimensions described and a possible value range. While wikis for web page creating and editing do not require 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 might find numerous use cases where a moderate degree of completeness is required, for example, a middle size department is involved in collaborative development of a business document with a semi-formal structure (e.g. contract, project plan, business blueprint ...) which may lead to exceptions and inaccuracy. [36]. 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 [37] ContentFlow [38] which is based on the well known OpenACS [39] Community framework, an approach is presented on linking 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 [40] [41].

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.

3.2 Supporting 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 with 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 contest's 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 who 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 to perform tasks in specified sequences. This approach does not fit well with the openness characterising social software and is also responsible for some of the well-known limiting factors of these systems, such as problematic exceptions management and limited adaptation to changes. Social software environments are a place of collective intelligence. IT should support the knowledge workers not enforce their behaviour. In this section, a social software environment is suggested, to support the 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 is required to be performed by the actors changes 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).

Since we want participants to be able to change the process at any given moment on the basis of their experience and of their knowledge (for this reason the term "organisational best practice" seems more appropriate than workflow in this context), a well-defined process model cannot be assumed. Not having a predefined process model, however, does not mean that there is not an idea about how the process should be enacted. What usually happens is that, after the first few iterations, the process "takes shape", becoming more and more structured. It is still possible to support the continuous evolution of a process and exception handling is much simpler. One important point to note is that missing a formalised process model does not mean missing a process state. A state concept is needed in order to be able to help users in replying to the 'who', 'what', 'when' questions. 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) is a standardised graphical process notation that is experiencing rapid adoption among BPM tools vendors. The classic BPM approach is to use a language like BPMN to model the process that has to be enacted. 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 approach consists of using a representation of the state of the process in the form of a collection of data and metadata (i.e., content and context). By extracting and aggregating all process-relevant information from the artefacts, it is possible to define the current state of the process. The obvious next step is to take advantage of this knowledge to support users in applying organisational best practices by suggesting to 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 organisational best practices.

Automation can be supported in a peer-to-peer fashion, for example in Social X-Folders [42]. 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 they are in charge of accomplishing when the process reaches a given state.

In Social X-Folders, aggregation of feeds (coming from various sources: forums, blogs, wikis, shared agendas, etc) is 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 [43].

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.

The resulting system plays nicely within a social software framework showing that BPM and social software can take advantage of one another. Also in the context of workflow enactment, shared information can be used to determine the state of the process, (shared) rules can be used to activate automatic task execution in a peer-to-peer fashion. Users do not have their tasks enforced but are supported by the systems which also give them the chance to ignore their suggestions and find better alternatives.

3.3 Drawbacks and Pitfalls

While the utilization of social software has shown potential in the enactment of business processes, there are also drawbacks and pitfalls.

Social software has proved to be particularly promising in supporting ill-defined (often knowledge intensive) processes which tend to require rich human collaboration. Therefore, making the social software solution attractive for participation by humans becomes an important issue. This forces organisations to stretch their thinking from functionality-based approaches towards considerations of sociality [44]. Furthermore, it has been claimed that utilising social software can be very labour intensive requiring commitment to continuous content generation and maintenance [45].

In general, social software needs a ramp-up phase in which to become useful. The value of social software results from the multitude of contributions, which need of course some time to be created. Due to the lack of hierarchic organisation, there is also no planning possible. Instead, the users and potential contributors need to be convinced of the advantages of social software.

Another point is the difficulty of evaluating the use of social software in the enterprise. Already the intra-unit use of information technology is very difficult to rate. Evaluating the creation of weak ties etc. is even more difficult, due to their scattered benefits. Therefore, the calculation of a

return-on-investment of social software initiatives is a very daunting task. However, social software has this problem in common with services that are also very difficult to evaluate.

In the next section a strategic view is taken to evaluate possible usages of social software within enterprises.

4 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.

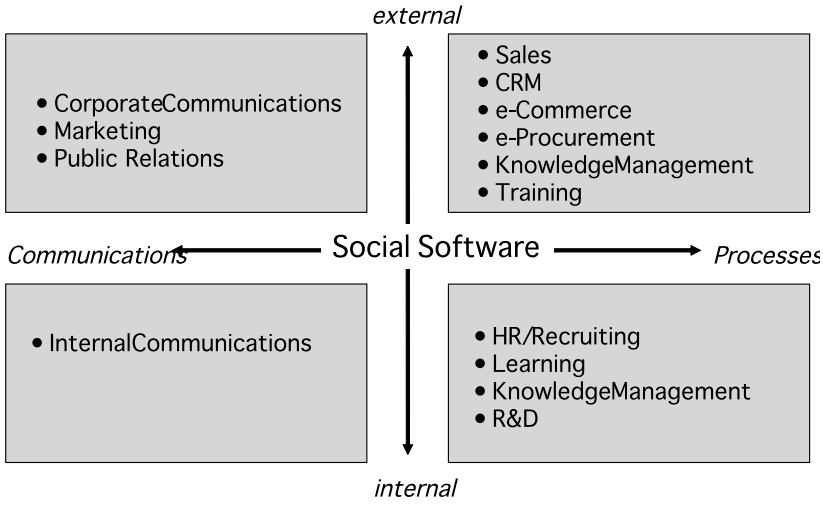

Figure 4: Social software influence in an enterprise [46]

The main effects of social software will be recognised in all knowledge related processes. These processes may include sales, marketing, innovation or human resources. Relevant questions are: 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? [47] [48]. In the next subsection some examples are described.

Some enterprises have gained a competitive advantage by having a community of innovative users connected with the enterprise's product [48] [49]. 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 [50] and innovations [51] 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 [52]. It has also been acknowledged that social software enables international reach,to help companies gain access to potential customers and co-developers all over the world [53]. Furthermore, some enterprises have begun to explore the idea of utilising social software to guide their product development [54]. 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” [49].

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 [47], the enterprises are likely to be forced to reconsider their business processes.

These challenges have led to proposing a dimensional view illustrating different utilisations of social software and their relationship to the enterprise's business processes (Figure 5) [55]. The first dimension, 'targeted stakeholders', describes for whom the particular instance of social software is intended. It has been 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. Differing objectives for social software have been identified, ranging from the support of product development related collaboration to supporting business-related aspects of the enterprise’s product. Based on these two dimensions, four categories have been formed and labelled with business processes they are likely to affect: 1) brand building, 2) distributed product development, 3) sales & partner support and 4) user involvement.

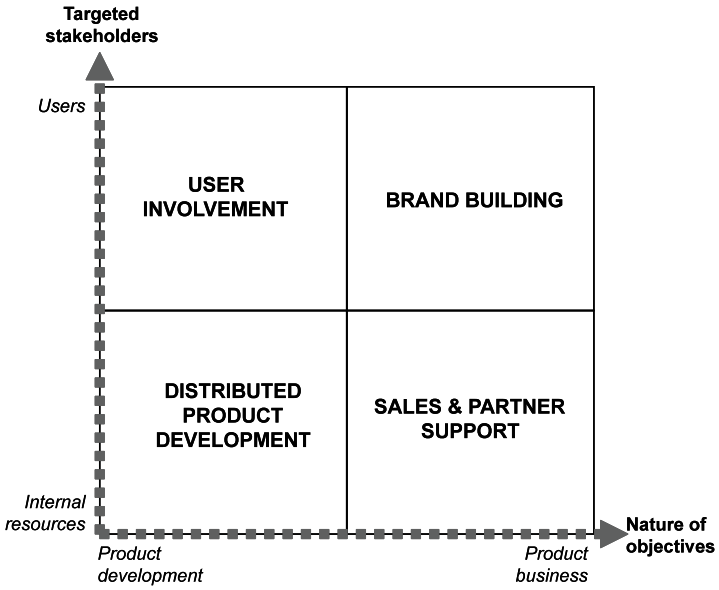

Figure 5: ***Dimensional view illustrating different utilisations of social software and their relationships to business processes***

The proposed dimensional view is valuable as 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 which 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 appear to have an effect on how the sociability of the intended solution should be supported, it is hoped 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.

The use of social software poses new requirements concerning digital identity and reputation. They are analysed in the following section.

5 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 also to create mechanisms for the human agents within the enterprise to add value to the knowledge of the company [56]. The Enterprise 2.0 term [8] [57] is gaining adoption.

This section of the work will first look at some of the overarching concerns of digital identity in an enterprise context. A more detailed technical machine learning and human agent driven approach will be outlined. In the last subsection, conceptual issues behind the nature of trust and reputation will be addressed which will assist in framing the problem space in this area.

5.1 Enterprise and Identity

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 both to analyse the existing data sets and the data for potential pseudonyms and to present a digital identity resource from which to make human agent based assertions. In section 2.4, 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 [58]. Studies have been performed that show if people are subjected to an overly dense information stream [59], 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 the less important.

5.2 Unified Digital Identity Resource

The next subsection will present a scenario being explored currently, which will help to put the concept of a programmatically available digital identity resource in context. A resource in this domain may be viewed as a component of a larger enterprise tool. This component could, on

demand, provide a contextually relevant result from an identity fragment, or partial identity artefact. The resultant data would give the broader context of an individual human agent within the bounds of a specific enterprise.

By providing a programmatically accessible digital identity resource via a REST based JSON/XML api, a social software system will enable analysis of the interconnected nature of the human agents acting within the system and the data which they generate. By layering human agent generated data on top of existing data, new forms of interconnectedness can be mined. This interconnected data on human agent activity will, in turn, enable more lightweight ad hoc business processes to be executed in a more effective manner. Rather than adopting an abstraction of the human agent within a system, via worklists or pattern identification [60], the system, called Reputation-based Message Routing, takes an alternate approach by providing human agent recommendations based on such analysis gained from data artefacts such as mailing lists, commit logs and bug ticketing data [61] down to the granularity of a specific human agent.

A simplified example of such analysis can be seen in the diagram below (Figure 6). An email thread and a bug report have two common human agents (agent one and two). From this initial inferred connection, the system can deduce that a second email thread, which also has agent two as a participant, might be of interest to agent one. For a real use case, there would need to be many more inferred connections of commonality before such a recommendation would be made, but in the small, such an aggregation of content and agents across data sets can show interesting results. This is only possible once a unified digital identity resource is in place.

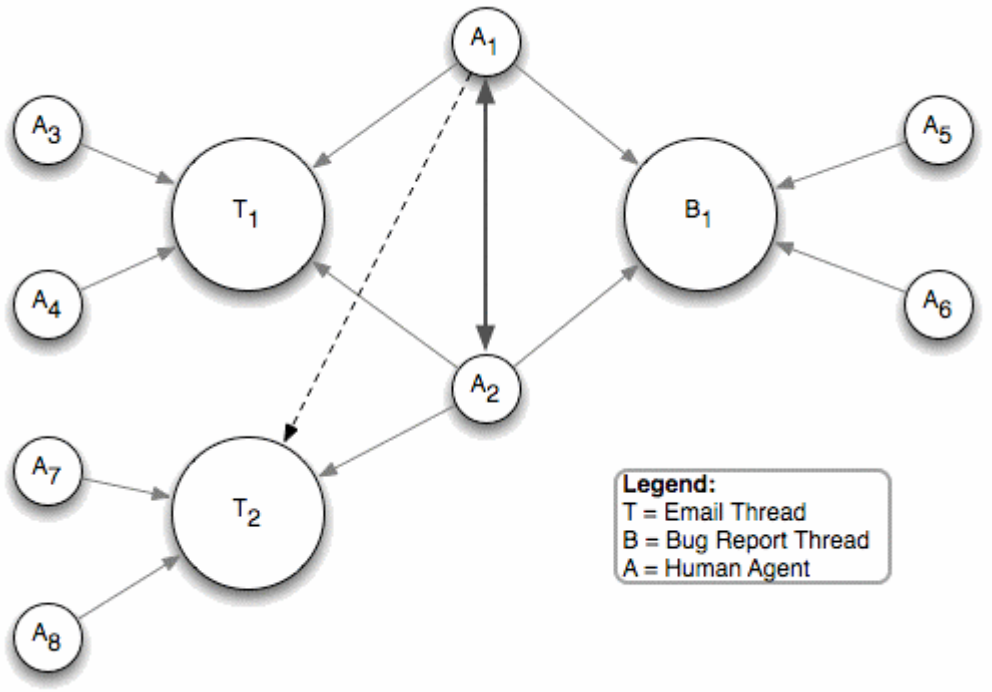

Figure 6: Thread, Bug and Human Agents Connectivity

As well as a fully automated approach to the identification of a specific human agent within the enterprise, this approach can also be extended by enabling a lightweight mechanism for facilitating human agents to correct and extend the automated identity recognition. By letting people tag people [62], in conjunction with the machine learning oriented approach, a significantly richer source of identity is possible as well as using human intelligence and expertise to correct and extend the automated identity recognition. By leveraging existing assets, this approach will

solve the adoption problem and bring a greater value to the adoption of social software within the enterprise as well as providing contextual meaning to the relationship between people and between data and its creators.

5.3 Trust and Reputation

There is now a notion of the requirement for digital identity in the enterprise and an overview of a technical approach that would provide such and resource. In this last subsection, the very nature of trust and reputation in a digital domain will be addressed.

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 [63]. 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 [64], 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 [65]. This measure may alleviate concerns of opportunistic behaviour [66] from the other participant in a given transaction. This opinion led abstraction of a deficit of information [67] can form the foundation of a decision making process. The act of aggregating an individual user's interpretation of ambiguous or asymmetric knowledge [68] 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 reputational resource. 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.

6 Conclusion

Combining business process management and social software offers a number of benefits. Social software allows the integration of 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 of openness and ease of usage are the pillars of the wide acceptance of social software seen mainly in the private sector. Therefore, proposing similar effects in the business environment means changing communication principles from predefined, hierarchic communication structures. Furthermore, the divide between abstract process

models, lifecycles, evaluations and the executed processes, can be narrowed or even avoided completely. 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 and 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 which may help to achieve an individual modelling goal.

Wiki enabled workflow systems seem 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 bottom-up manner. Content creators are not predefined, each user may add context – by tagging, evaluating, commenting or even reading. The sum of all these interactions is a new content in 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 upon 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.

7 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.

[3] M. Granovetter, “The Strength of Weak Ties,” *The American Journal of Sociology*, vol. 78, 1973, pp. 1360-1380.

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

[5] A. Majchrzak, C. Wagner, and D. Yates, “Corporate wiki users: results of a survey,” *Proceedings of the 2006 international symposium on Wikis*, Odense, Denmark: ACM, 2006, pp. 99-104.

[6] A. Kittur and R.E. Kraut, “Harnessing the wisdom of crowds in wikipedia: quality through coordination,” *Proceedings of the ACM 2008 conference on Computer supported cooperative work*, San Diego, CA, USA: ACM, 2008, pp. 37-46.

[7] F.B. Viegas, M. Wattenberg, J. Kriss, and F. Van Ham, “Talk before you type: Coordination in Wikipedia,” *Hawaii International Conference on System Sciences*, IEEE, 2007, p. 1298.

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

[9] P. Dadam, “The Future of BPM: Flying with the Eagles or Scratching with the Chickens?,” *Business Process Management*, 2008, p. 3.

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

[11] Y. Benkler, *The Wealth of Networks : How Social Production Transforms Markets and Freedom*, Yale University Press, 2006.

[12] 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.

[13] 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, pp. 243-258.

[14] 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.

[15] 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.

[16] M. Granitzer, M. Kröll, 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)*, IEEE Computer Society Press, 2008.

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

[18] 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.

[19] T. Hornung, A. Koschmider, and G. Lausen, "Recommendation Based Process Modeling Support: Method and User Experience," *Proceedings of the 27th International Conference on Conceptual Modeling, Barcelona, Spain, October, 2008*, Springer, 2008, pp. 265-278.

[20] A. Koschmider, M. Song, and H.A. Reijers, "Social Software for Modeling Business Processes," *Proceedings BPM 2008 Workshops, LNBIB*, Heidelberg: Springer, 2008, pp. 642-653.

[21] M. Song, "Organizational mining in Business Process Management," PhD thesis, Pohang University of Science and Technology, Pohang, South Korea, 2006.

[22] A. Koschmider, M. Song, and H.A. Reijers, "Advanced Social Features in a Recommendation System for Process Modeling," *12th International Conference on Business Information Systems*, Poznan: Springer, 2009, pp. 109-120.

[23] Technorati, "Sifry's Alerts: State of the Blogosphere, February 2006 Part 1: On Blogosphere Growth."

[24] Wikipedia, "Statistics - Wikipedia, the free encyclopedia."

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

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

[27] 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, pp. 378-378.

[28] 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 2007, Trondheim, Norway*, 2007.

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

[30] 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.

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

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

[33] 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.

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

[35] 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.

[36] 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.

[37] G. Neumann, "XoWiki - Towards a Generic Tool for Web 2.0 Applications and So-cial Software," Vienna: 2007.

[38] G. Neumann and S. Erol, "From a social wiki to a social workflow system," *Proceedings of the BPM 2008 Workshops*, Heidelberg: Springer, 2008.

[39] "OpenACS Home."

[40] 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.

[41] 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.

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

[43] G. Leshed, E.M. Haber, T. Matthews, and T. Lau, "CoScripter: automating & sharing how-to knowledge in the enterprise," *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, Florence, Italy: ACM, 2008, pp. 1719-1728.

[44] W. Bouman, T. Hoogenboom, R. Jansen, M. Schoondorp, B. de Bruin, and A. Huizing, "The realm of sociality: Notes on the design of social software," *ICIS 2007 Proceedings*, p. 154.

[45] T. Lehtimäki, J. Salo, H. Hiltula, and M. Lankinen, *Harnessing Web 2.0 for business to business marketing- literature review and an empirical perspective from Finland*, Oulu University Press: Faculty of Economics and Business Administration, University of Oulu, 2009.

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

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

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

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

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

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

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

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

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

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

[56] B. Jennings and A. Finkelstein, “Digital Identity and Reputation in the Context of a Bounded Social Ecosystem,” *BPM2008 Workshop Proceedings, Springer – LNCS, Milano, 2008.*

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

[58] 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.

[59] 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.

[60] N. Russell, A.H.M. Ter Hofstede, D. Edmond, and W.M.P. van der Aalst, *Workflow resource patterns*, Beta, Research School for Operations Management and Logistics, 2005.

[61] B. Jennings and A. Finkelstein, “Flexible Workflows: Reputation-based Message Routing,” *Proceedings of BPMDS*, vol. 8, p. 97.

[62] S. Farrell and T. Lau, “Fringe contacts: People-tagging for the enterprise,” *Collab. Web Tagging Workshop in conj. with WWW2006*.

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

[64] 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.

[65] 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.

[66] 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.

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

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