

Emergent Case Management for Ad-hoc Processes: A Solution Based on Microblogging and Activity Streams

Martin Böhringer

Chemnitz University of Technology
Thüringer Weg 7, 09126 Chemnitz, Germany
martin.boehringer@wirtschaft.tu-chemnitz.de

Abstract. Recent research has shown the need to include unstructured ad-hoc processes into business process management. A possible solution for this purpose is Case Management, where information related to a certain process instance is bundled into a case file. In addition to existing top-down approaches, this paper suggests a bottom-up view on Case Management that leverages emergent user-driven case handling. We theoretically derive characteristics of such a system and demonstrate the approach based on a toolset of current Social Software techniques including microblogging, activity streams and tagging.

Keywords: Ad-hoc processes, Social Software, Microblogging, Activity Streams, Tagging, Case Management, Process-enhanced groupware, People-intensive processes.

1 Introduction

Many business processes are difficult to model [19]. That is especially true for knowledge-intensive tasks such as those found in incident management, consulting or sales. The reason for the impossibility of modeling can often be found in the ad-hoc characteristics of these processes. Owing to different contexts and fast-changing environments, they cannot be determined at an appropriate level of granularity before a process begins. However, to achieve the holistic management of all business processes, business process management (BPM) cannot only focus on well-structured, easy-to-model processes but has to integrate ad-hoc processes too. Accordingly, Dustdar et al. [8] compared BPM's classical focus by emphasizing "low hanging fruits," whereas most processes would not fit that pattern and might need different support.

Case Management represents a promising approach to support ad-hoc processes, because it accepts that these activities cannot be modeled in advance and, therefore, provides the minimum amount of documentation and standardization compared with widely used tools such as email and phone. Owing to the increasing importance of ad-hoc processes, this concept has seen a revival in current vendor products and practitioner discussions (see, for example, [7,17,22]). Although existing approaches for Case Management provide more degrees of freedom for process executers and support unstructured processes, they still have a top-down nature and create additional work for their users. Therefore, user adoption can be questioned. This paper suggests

a bottom-up view of Case Management which enables emergent case creation and structuring. It argues that such an approach has to focus on process actions (e.g. activities in information systems such as ERP) and artifacts (e.g. information snippets) and, therefore, would have to take into account software tools that are already being used by process executors. Based on a conceptual prototype we argue that current developments in Social Software (i.e. tagging, microblogging and activity streams) especially show a perfect fit to build Emergent Case Management systems that seamlessly integrate into everyday work routines.

After theoretically explaining this approach and discussing the existing literature, we develop a framework for Emergent Case Management in Chapter 3. We theoretically derive requirements towards such a system and argue that these requirements can be found in the Social Software principles discussed in Chapter 4. Finally, we demonstrate a prototype based on microblogging, activity streams and tagging in Chapter 5 before discussing our results. A conclusion ends the paper.

2 Ad-hoc Processes

Traditionally, the research and practice of business information systems focuses on well-structured business processes. We have a rich tradition and broad body of knowledge to model these processes and build information systems based on them. However, there are an increasing number of publications both from academia and practitioners arguing that this focus might be too narrow (see [3,12,19,31]). In fact, many activities are started and conducted in an ad-hoc way rather than being planned in advance [31]. This is especially the case for knowledge-intensive or project-based activities [14], which often represent the core competencies of an organization [19].

Bernstein [3] identified the two poles of “highly unspecified” and “highly specified” processes (Figure 1). Although the latter is the classic target of process-based information systems, the first is mostly discussed in disciplines such as computer-supported cooperative work or, more recently, Social Software. Processes belonging to this area including much of the space between the two extremes have been discussed as “ad-hoc processes.” They represent a significant part of enterprise activities and, therefore, are an important topic of interest. Surprisingly, compared with its importance, this area has been covered little in previous research.

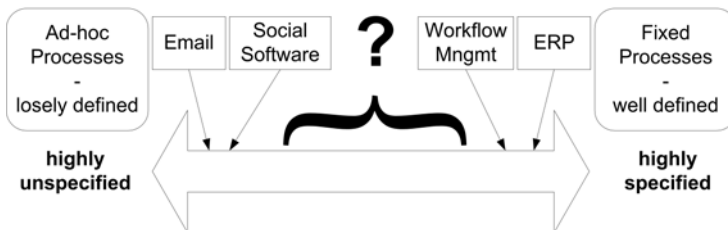


Fig. 1. Degree of process specification (modified from [3])

Ad-hoc processes are sets of business activities and corresponding artifacts (e.g. information, decisions and products) that can only be standardized at a high level of aggregation. The actual kinds of activities and their ordering are different from case to case. Chung et al. stated that “while certain characteristics and activities can be predicted, much of the process cannot be fully specified at the start, since it requires information that only becomes available some way into the project” [6]. If we assume that in the context of ad-hoc processes the next step is never determined, they can only be defined entirely in the form of their instantiations *after* they have been finished. This means that “ad-hoc processes [...] have no underlying process definition” [8]. Therefore, ad-hoc processes by definition cannot be standardized and as such their execution cannot be controlled by classical process-based information systems. This means that human beings and, in most cases, knowledge workers are in control of the process. Jennings and Finkelstein argued that where the interaction of human agents is required, a top-down approach is not suitable because of the lack of *a priori* knowledge [14]. Besides the fact that it seems impossible to think of all possibilities by design time, such a process model would become complex and unmanageable [34].

3 Emergent Case Management

Case Management (also referred to as Case Handling or Case Processes) is an approach to handle business activities that cannot be supported by classical BPM tools. It focuses on process instances and is a collaborative and communicative task [7]. Its origin is situated in domains such as healthcare and governance [17] and, therefore, can be considered to support especially knowledge-intensive processes [34]. Forrester analysts Le Clair and Moore [17] suggested the increasing importance of the concept and OMG (Object Management Group, <http://omg.org/>) is working on an according standard [22]. However, the exact meaning of Case Management in terms of its actual implementations is still vague and depends on context and use. Therefore, it can be seen as a general approach of handling unstructured business processes rather than being a mature tool category. Accordingly, our general understanding of Case Management in terms of a working definition is as follows: *Case Management is an approach to handling ad-hoc processes. It supports process instances (cases) in providing a collaborative space to store and negotiate case-related activities and artifacts (e.g. information, decisions, workflows) without the need for the ex ante modeling of the whole process.*

Existing Case Management approaches nonetheless are oriented towards top-down organization and pre-configuration. Van der Aalst et al. [34], for example, proposed a system that supports the user with information about what activities can be performed based on the current status of the case. This suggests that the Case Management system has to be trained and possible case statuses have to be modeled in advance. Although useful in contexts that are at least semi-structured, this might be a too rigid approach for ad-hoc processes. If we take into account that ad-hoc processes cannot be modeled in advance, a top-down pre-configuration of possible case states might also be incomplete if not impossible. Le Clair and Moore [17] explained Case Management in the

context of “people-driven processes.” If users are drivers of ad-hoc processes and are crucial to their successful execution, it might be worthwhile thinking about a paradigm shift towards bottom-up solutions. Hagel and Brown [11] suggested that shifting from a push-based approach in BPM and information systems in general to a pull-based, user-driven model is a logical consequence of rising complexity and knowledge-intensity. Applied to Case Management this leads us to the proposal of Emergent Case Management.

Emergent Case Management is an approach for the bottom-up managing of ad-hoc processes. Although existing concepts including “classical” Case Management try to leverage the benefits of structured BPM for unstructured cases and, therefore, modify its methods to achieve greater flexibility, we suggest starting at concepts for unstructured processes (e.g. email and Social Software, Figure 1) and aiming towards enabling users to organize this content in a process-based view. Therefore, people are at the center of such an information system; they are in charge of the process. They have to be able to adopt the tool for their process-related needs. Since the goal is to integrate as many case-related users at possible (executors, stakeholders, knowledge carriers), such an approach has to be as simple as possible and should not burden them with restrictions [26].

Figure 2 visualizes the underlying framework and understanding of Emergent Case Management. Users are at the centre of the solution and the driving force of all actions. In working on an ad-hoc process, they execute activities and act on artifacts such as information, decisions and documents (e.g. by creating, executing and working on them). These artifacts are represented in information systems (including non-technical systems such as face-to-face conversations). The goal of Emergent Case Management is to enable users to assign activities and artifacts independent of their representation to a certain case, which can be dynamically defined by users.

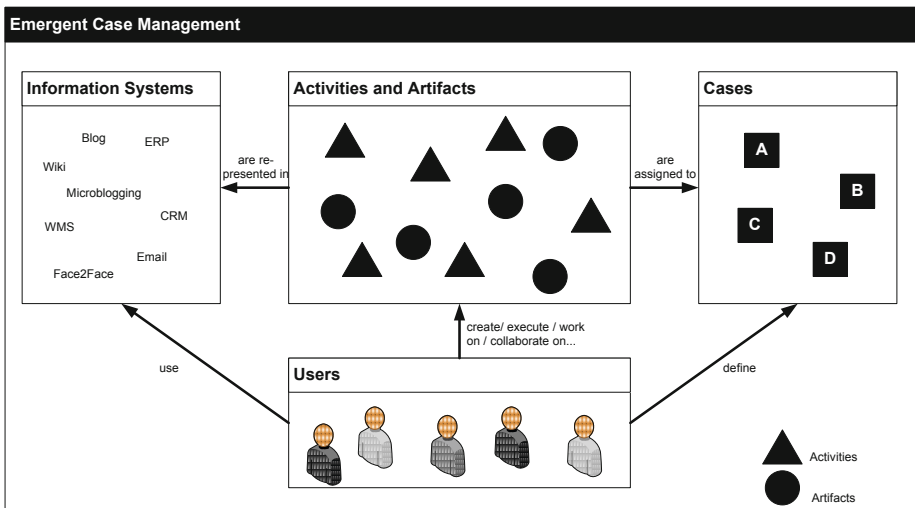


Fig. 2. Conceptual schemata of Emergent Case Management

4 Emergent Case Management and Social Software

There is an ongoing discussion about how BPM and Social Software fit together. Approaches include a wide range of use cases, such as supporting process documentation using wikis, collaborative process modeling and the deployment of processes [9]. Traditional BPM and Social Software leverage different approaches to support users. Johannesson et al. [15] highlighted the points “external authority vs. voluntary participation,” “definite endpoint vs. open-endedness,” “major efforts vs. quick contributions” and “access control vs. transparency.” Owing to these characteristics, several authors have argued that ad-hoc processes show many parallels to Social Software characteristics and have suggested that these tools support unstructured processes (see, for example, [3,15,20]).

A distinct principle of most Social Software tools is their design as a platform [23] that enables but does not force social interaction: “sociality cannot be designed; it can only be designed for” (Wenger [37] paraphrased in [13]). Although classical IT applications are designed for a special purpose and with a special use or business process in mind, platforms go one step back and acknowledge that only the users and process owners in their special (ad-hoc) situations can specify these use cases. Instead of trying to support an ideal process, platforms support users with basic functionalities and a high degree of freedom. The goal of these systems is that users can leverage them to create ad-hoc suitable IT support for their tasks [15].

Based on the framework (Figure 2) and the previous discussion, our assumption here is that the focus of ad-hoc process executers is on getting things done. They are involved in several parallel cases, and this includes a massive amount of multitasking. Since ad-hoc processes are seldom repeated, for workers in ad-hoc processes it is important that the goal can be reached rather than *how* this has to be done [31]. Therefore, the flow of activities is especially crucial for their work. This is why Social Software tools supporting a flow-based representation of their work are central to the concept of Emergent Case Management. A typical platform for this task is microblogging (e.g. Twitter), where users can publish small pieces of information and enhance it with structure (e.g. hashtags “#bpm2010”, re-tweets, @-mentions). Combined with automatically generated activity data this leads to so-called activity streams, which can be part of a separate application (e.g. Facebook’s “News Feed”) or included in microblogging (described, for example, in the concept of “ubiquitous microblogging” [4]).

In a case study on the long-term usage of a microblogging-like enterprise software system called Arinia, Barnes et al. found that Arinia is used to support nearly all customer-related processes [1]. In this case, these processes are project-driven and, therefore, highly unstructured. The company is certified for ISO 9001 and Arinia is seen as a key concept for supporting their processes. Following this initial insight, we suggest that it is blogging, especially microblogging, that is of great interest to BPM, as it follows an activity-driven, time-based approach of interaction management. Rosemann argued that its potential for BPM is still “under-utilized” [28]. Other researchers and practitioners support the idea of microblogging-based process execution (see, for example, [16,35]) and the first vendors are starting to integrate similar features into their products [32]. Furthermore, the approach of focusing on activities is in line with previous research on activity-centric collaboration [10].

5 A Prototype of Emergent Case Management

5.1 Conceptual Description

A basic assumption of the concept of Emergent Case Management is that modeling is not applicable for ad-hoc processes. We argue that ad-hoc processes cannot be modeled by definition and that modeling a single process instance during execution would only create additional work but no benefits. Therefore, we aim to provide bottom-up process support by enabling users to assign artifacts and activities to a case (that is, a process instance). We suggest using activity streams of automatically, semi-automatically and manually created activity logs (microblogging postings), where artifacts are represented a) by activities executed on them (e.g. “Max created the document ‘Requirements Engineering’”) or b) in postings directly referring to them (e.g. via linking). Table 2 contains examples of these information sources.

Table 1. Sources of activity logs

Source	Description and examples
Manual creation	Since people are in charge of ad-hoc processes, manually created activity logs using microblogging are the focus of our concept. Previous research has suggested that in leveraging microblogging, half of intraorganizational communication can shift from emails to this open communication medium [1], which should be a major goal because email and face-to-face communication cannot be included in the activity stream.
Semi-automated creation	Lots of web applications include a “Tweet this” button, which allows users to edit and tweet a pre-generated text. After finishing rescheduling in the production planning system, user Max could use this functionality to generate the text, add the appropriate hashtags and maybe a certain user who waits for this input to proceed and finally send the posting to the activity stream: Example: Max: “@Paul I created ‘Production Planning Rescheduled’ at http://bit.ly/2nkmNd #repabc #finished”.
Automated creation	Activity logs are automatically created from existing information systems such as CRM, ERP or wikis when actions are conducted. Where possible, such structured tools should be used for parts of the ad-hoc process and, therefore, be integrated into the activity stream [26]. Assignment to a case can be achieved in two ways: a) the source information system ‘knows’ the case hashtag and includes it in the post (requires a certain level of standardization) or b) users re-tweet such activity logs if they are important for the case and add appropriate hashtags. Example: SAP: “Paul approved order O123.” → Max: “RT @SAP Paul approved order O123. #repabc”

Furthermore, Figure 3 presents an overview of the framework. Based on the central understanding of Emergent Case Management, the goal is to include as much information into the activity stream as possible. Therefore, such a system should not ignore existing systems. Where possible, specialized information systems such as ERP or CRM should be used during process execution. The so-created activities and artifacts are posted into the activity stream where they can be assigned to a certain case. As is known from microblogging, the stream is ordered chronologically with the newest entries on top. We propose using hashtags for assigning activity entries to cases. The principle is well known from Twitter and provides high degrees of freedom for users. We will discuss the pros and cons of this decision in Chapter 6.

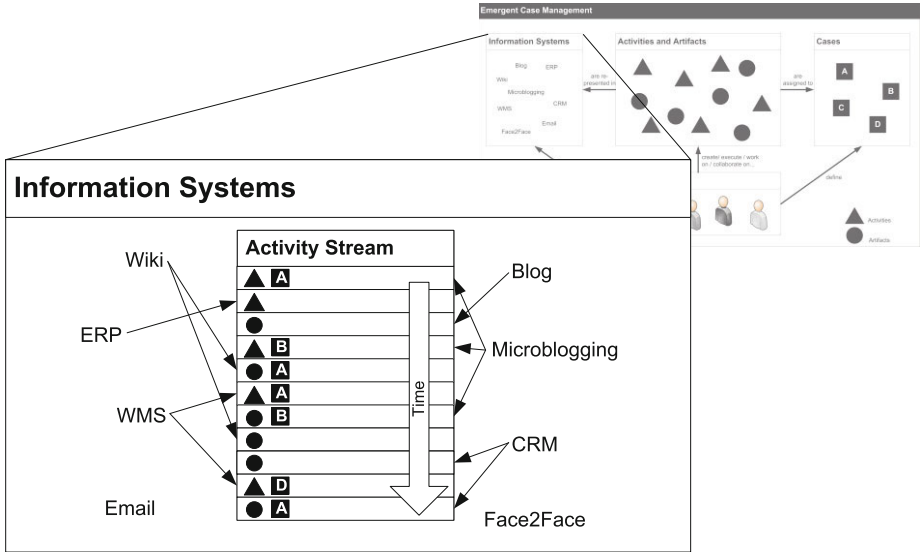


Fig. 3. The prototype’s concept of activity stream-based Emergent Case Management

In applying microblogging/activity streams combined with tagging, we are able to support all components of our Emergent Case Management framework. Table 2 summarizes the representation of the different concepts.

Table 2. Concepts of Emergent Case Management and their representation in the prototype

Concept	Representation of the concept
Users	Participants in microblogging, using microblogging for communication
Activities	Microblogging postings and activity stream entries
Artifacts	Links in microblogging postings/activity stream entries
Legacy IS	Own microblogs or postings to the user’s activity stream
Cases	Hashtags (e.g. “#emergency_test”)
Assignment to cases	Using hashtags in a posting (e.g. “Reminder: the #emergency_test is scheduled for tomorrow.”)

5.2 Front-End

To visualize the concept of Social Software-based Emergent Case Management, we will use a short example based on a real world case to demonstrate our prototype (for a comprehensive discussion of the case, see [5]). Our demo company is in the textile business and produces high quality test series. Projects are unique and there are little possibilities for standardization. Customers might require short-run changes to existing or sometimes even already shipped orders. These time-critical ad-hoc processes are called “reparations.” Our example refers to such a set of activities for customer “ABC.” That is why the sales representative started the process instance with the hashtag “#repabc”.

To visualize the prototype we use a Twitter-like interface. Process-related stakeholders as well as existing information systems are represented via their own microblogs, which are aggregated in a joint activity stream. For the manual creation of activity entries, people have a simple text box restricted to 140 characters. In our example, a search for the hashtag “#repabc” shows the complete ad-hoc process documentation including human and non-human activity, artifacts (e.g. the bill) and parts of the workflow (using @-referencing), as shown in Figure 4. The Twitter-inspired search screen also shows additional possibilities for analytical support (e.g. trending topics at the right side). Combined with text mining techniques such as opinion mining, such scenarios can enable constant process monitoring and process intelligence (see, for example, the feature “Social Business Intelligence” of enterprise microblogging vendor socialcast.com).

Realtime results for #repabc

katrinmueller RT @SAP_Receipts bill no. #R23992383 has been paid by customer #ABC #repabc
about 6 minutes ago from web

thorstenmann Quality checked, order shipped to customer #repabc
2 days ago from web

ottomuster #repabc forwarded to @thorstenmann for quality check.
2 days ago from web

thomu Meeting with @ottomuster on #repabc
3 days ago from web

thomu RT @ext_klein (Ms Klein, #ABC): Timely delivery is of highest priority for us! #repabc
4 days ago from web

thomu On my way to ##Dresden visiting @ext_klein of #ABC #repabc
4 days ago from mobile client

ottomuster RT @Machine1839 Error Code #F2328. Actuator broken. //@thomu #repabc will be delayed minimum 1 day!
4 days ago from web

ottomuster @thomu we will take care of it, probably will be finished tomorrow appr. 2pm. #repabc
5 days ago from web

thomu @ottomuster Customer #ABC requests repairing the current delivery: the color fidelity is insufficient (cf. colour sample ABC12) #repabc
5 days ago from web

Trending topics

- #repccc
- quote
- #repabc
- customer
- #annualreport
- #fail
- #canoetrip

Fig. 4. Monitoring of ad-hoc processes via hashtag filtering

5.3 Technological Foundation

As the scope of this paper is to present the general concept of using microblogging/activity streaming for supporting ad-hoc processes, we will only briefly discuss

the technological foundation of an appropriate solution. We have evaluated the approach in different laboratory experiments with several existing microblogging applications such as Twitter and StatusNet. Although they show good support for manually created microblogging postings, for automated and semi-automated activity reporting a more sophisticated solution is needed. Big Internet vendors such as Facebook, MySpace and Opera are currently cooperating on a standard format for activity exchange (activitystrea.ms) based on the Atom feed protocol. We suggest building adapters for existing information systems such as CRM and ERP, which publish activity information from these sources in an activity streams format. Our back-end system, therefore, consists of a) a communication and extraction layer to retrieve this information, b) a storage layer and c) a representation layer providing the information feeds to d) the front-end (where users can subscribe/unsubscribe and publish information). Until we have implemented such a system, Twitter, with its millions of users, might be a good proof for the scalability of the concept.

6 Discussion

Our approach of using hashtags as case identifiers provides several possibilities for enhancing structure and standardization. As it can be seen with Twitter, hashtags at first are emergent terms that can be freely defined by users. As observed especially for events such as conferences, different hashtags often evolve for the same event (e.g. “bpm”, “bpm10” and “bpm2010”), which leads to fragmented information. A way of handling such problems could be to provide predefined tags or the predefined structuring of tags. However, this is top-down thinking and might decrease people’s motivation to use the system. From previous research we already have some knowledge about emergence and shared vocabulary building in folksonomy systems. Muller reported on four tag-based systems in an enterprise and found that tagging use is only consistent inside one system [21]. Interestingly, people used different tags in different systems, which provides evidence for users’ possibilities of adapting certain rules in special contexts. Other researchers found similar patterns and described folksonomy building as a “negotiated process of users” [24] and a “self-organizing system towards a shared vocabulary building” [18], which develops towards a group consensus. Therefore, a suitable bottom-up mechanism for governance would not be to restrict tag usage but to enable documentation (e.g. via hashtag wikis similar to hashtags.org or hashdictionary.com) or ways of defining synonymous tags.

Another critical point of the proposed concept is dealing with information overflow. In particular, their context is important to support a flexible process [27]. Therefore, all available information should be provided for process stakeholders. However, this might lead to a plethora of fine-granular activity entries. If we think of the typical knowledge worker

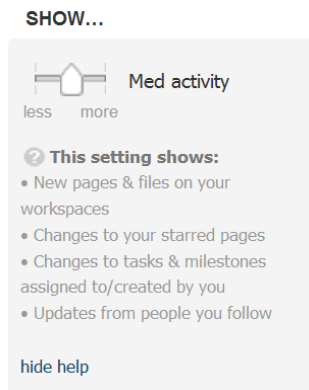


Fig. 5. Activity slider at pbworks.com

as being involved in a high number of parallel ad-hoc processes, there is a need for suitable mechanisms to avoid information overload. A possible solution is to leverage rich user interfaces. Figure 5 shows the activity slider of the Social Software suite PBworks. The user intuitively can filter the incoming information flow by its importance (with human messages being most important and automated activity logs being least important).

Owing to its platform characteristics, the presented solution for Emergent Case Management is expected to be strong in enabling creativity, which can be seen as a key element in ad-hoc processes [30], and facilitating related effects such as serendipity [36]. However, further work has to focus on the possibilities for process evolution. Well-structured business processes can evolve from ad-hoc activities [2]. If a certain process stabilizes or a case has to be repeated, techniques such as process mining [33] could be applied to the activity log to distillate the process structure. Therefore, a strong technological foundation in terms of semantics and open standards is necessary. Besides current initiatives such as the standardization of activity exchange there are already a number of works on reliable decentralized and semantically enriched microblogging solutions (see, for example, [25,29]) that could be applied to the presented scenario.

7 Conclusions

The paper's contributions are as follows: a) it presented a bottom-up solution for supporting and managing ad-hoc processes called Emergent Case Management; b) it suggested using Social Software, in particular a combination of cutting edge technologies such as tagging, microblogging and activity streams, for Emergent Case Management; and c) it presented a conceptual prototype related to a real world example that showed the applicability of the approach. The focus of the discussed approach is on its integration into everyday work and on it being as lightweight as possible. The management part, therefore, is reduced to creating organizational foundations (e.g. simple rules for hashtag creation), gardening the folksonomy and supporting evolving process structures. This represents a paradigm shift in BPM: instead of supporting central planners and analysts, our focus for supporting ad-hoc processes is primarily on the process executers themselves. Hence, factors such as the joy of usage and simplicity become important.

Current vendor developments point towards Emergent Case Management. Salesforce has announced Salesforce Chatter, an activity streaming module for its enterprise software suite. Microsoft Sharepoint will feature activity streaming in its 2010 version. Big Internet vendors such as Facebook, MySpace and Opera are cooperating to build a standard format for activity exchange (activitystrea.ms). In conclusion, the infrastructure for the proposed solution for ad-hoc process support is growing, and further works on the topic are needed to uncover this development's potential for BPM. Accordingly, our roadmap includes running an Emergent Case Management system and evaluating it in real world cases.

References

1. Barnes, S.J., Böhringer, M., Kurze, C., Stietzel, J.: Towards an understanding of Social Software: the case of Arinia. In: Proceedings of the 43rd Hawaii International Conference on System Sciences (HICSS-43). IEEE Computer Society, Los Alamitos (2010)
2. Bernstein, A., Schucan, C.: Document and process transformation during the product life cycle. In: Wakayame, T., Kannapan, S., Khoong, C.M., Navathe, S., Yates, J. (eds.) *Information and Process Integration in Enterprises - Rethinking Documents*, pp. 277–284. Kluwer Academic Publishers, Dordrecht (1998)
3. Bernstein, A.: How can cooperative work tools support dynamic group process? Bridging the specificity frontier. In: Proceedings of the 2000 ACM Conference on Computer Supported Cooperative Work - CSCW 2000, pp. 279–288. ACM Press, New York (2000)
4. Böhringer, M., Gluchowski, P.: The Beauty of Simplicity: Ubiquitous Microblogging in the Enterprise. In: Proceedings of the 12th International Conference on Enterprise Information Systems (2010)
5. Böhringer, M., Jentsch, D.: Lightweight IT support for ad-hoc-processes in production and logistics. In: Proceedings of the 43rd CIRP International Conference on Manufacturing Systems (2010)
6. Chung, P., Cheung, L., Stader, J., Jarvis, P., Moore, J., Macintosh, A.: Knowledge-based process management – an approach to handling adaptive workflow. *Knowledge-Based Systems* 16(3), 149–160 (2003)
7. De Man, H.: Case Management: A Review of Modeling Approaches. *BPTrends* (2009)
8. Dustdar, S., Hoffmann, T., Van der Aalst, W.: Mining of ad-hoc Business Processes with TeamLog. *Data and Knowledge Engineering* 55(2), 129–158 (2005)
9. Erol, S., Granitzer, M., Happ, S., Jantunen, S., Jennings, B., Koschmider, A., Nurcan, S., Rossi, D., Schmidt, R.: Combining BPM and Social Software: Contradiction or Chance? *Software Process: Improvement and Practice Journal*, Special Issue on BPM 2008 Selected Workshop Papers (2009)
10. Geyer, W., Muller, M., Moore, M., Wilcox, E., Cheng, L., Brownholtz, B., Hill, C.R., Millen, D.R.: Activity Explorer: Activity-centric collaboration from research to product. *IBM Systems Journal* 45(4), 713–738 (2006)
11. Hagel, J., Brown, J.: From push to pull – emerging models for mobilizing resources. *Journal of Service Science*, Third Quarter (2008)
12. Hall, J.M., Johnson, M.E.: When Should a Process Be Art, Not Science. *Harvard Business Review* 87(3), 58–65 (2009)
13. Hoogenboom, T., Schoondorp, M.: The Realm of Sociality: Notes on the Design of Social Software. *Working Papers on Information Systems* 8(1) (2008)
14. Jennings, B., Finkelstein, A.: Micro Work Gestural Analysis: Representation in Social Business Processes. In: Proceedings of the 2nd Workshop on Business Process Management and Social Software (2009)
15. Johannesson, P., Andersson, B., Wohed, P.: Business Process Management with Social Software Systems – A New Paradigm for Work Organisation. In: Proceedings of the 1st Workshop on Business Process Management and Social Software (2008)
16. Kemsley, S.: BPM, Collaboration and Social Networking #brf, Column 2 (2009), <http://www.column2.com/2009/11/bpm-collaboration-and-social-networking-brf/>
17. Le Clair, C., Moore, C.: Dynamic Case Management — An Old Idea Catches New Fire. *Forrester* (2009)

18. Maass, W., Kowatsch, T., Münster, T.: Vocabulary Patterns in Free-for-all Collaborative Indexing Systems. In: *International Workshop on Emergent Semantics and Ontology Evolution*, pp. 45–57 (2007)
19. Marjanovic, O.: Towards IS supported coordination in emergent business processes. *Business Process Management Journal* 11(5), 476–487 (2005)
20. Marjanovic, O.: Inside Agile Processes: A Practitioners Perspective. In: *Proceedings of the 42nd Hawaii International Conference on System Sciences (HICSS-42)*. IEEE Computer Society, Los Alamitos (2009)
21. Muller, M.J.: Comparing tagging vocabularies among four enterprise tag-based services. In: *Proceedings of the 2007 International ACM conference on Conference on Supporting Group Work - GROUP 2007* (2007)
22. OMG: Case Management Process Modeling (CMPM) - Request For Proposal (2009), <http://www.omg.org/cgi-bin/doc?bmi/09-09-23>
23. O'Reilly, T.: What Is Web 2.0 (2005), <http://www.oreilly.de/artikel/web20.html>
24. Paolillo, J., Penumathy, S.: The Social Structure of Tagging Internet Video on del.icio.us. In: *Proceedings of the 40th Hawaii International Conference on System Sciences (HICSS-40)*. IEEE Computer Society, Los Alamitos (2007)
25. Passant, A., Hastrup, T., Bojars, U., Breslin, J.: Microblogging: A Semantic and Distributed Approach. In: *Proceedings of the 4th Workshop on Scripting for the Semantic Web* (2008)
26. Rettig, C.: The trouble with enterprise software. *MIT Sloan Management Review* 49(1) (2007)
27. Rosemann, M., Recker, J., Flender, C.: Contextualisation of business processes. *International Journal of Business Process Integration and Management* 3(1), 47–60 (2008)
28. Rosemann, M.: Where BPM and Twitter could meet. ARIS Community (2009), <http://www.ariscommunity.com/users/mrosemann/2009-08-07-where-bpm-and-twitter-could-meet>
29. Sandler, D.R., Wallach, D.S.: Birds of a FETHR: Open, Decentralized Micropublishing. In: *8th International Workshop on Peer-to-Peer Systems, IPTPS 2009* (2009)
30. Seidel, S., Rosemann, M., Becker, J.: How does Creativity Impact Business Processes? In: *Proceedings of the 16th European Conference on Information Systems* (2008)
31. Tochtermann, K., Reisinger, D., Granitzer, M., Lindstaedt, S.: Integrating ad hoc processes and standard processes in public administrations. In: *Proceedings of the OCG eGovernment Conference* (2006)
32. Ultimus, Inc.: Ultimus Enhances BPM Suite with Twitter Updates for Your Business Process (2009), <http://www.ultimus.com/Blog/bid/33745/Ultimus-Enhances-BPM-Suite-with-Twitter-Updates-for-Your-Business-Process>
33. Van der Aalst, W., Nikolov, A.: Mining E-Mail Messages: Uncovering Interaction Patterns and Processes using E-mail Logs. *International Journal of Intelligent Information Technologies* 4(3), 27–45 (2008)
34. Van der Aalst, W., Weske, M., Grünbauer, D.: Case handling: a new paradigm for business process support. *Data & Knowledge Engineering* 53(2), 129–162 (2005)
35. Venkat, J.: BPM and Twitter – Tweeting to BPM (2009), <http://www.bouncingthoughts.com/2009/08/bpm-and-twitter-tweeting-to-bpm/>
36. Vinoski, S.: Serendipitous reuse. *IEEE Internet Computing* 12(1), 84–87 (2008)
37. Wenger, E.: *Communities of practice: Learning, meaning and identity*. Cambridge University Press, Cambridge (1998)