M. Herczeg & M. C. Kindsmüller (Hrsg.): Mensch & Computer 2008: Viel Mehr Interaktion.
München: Oldenbourg Verlag, 2008, S. 67-76

Combining Web 2.0 and Collaboration Support Systems

Michael Prilla, Carsten Ritterskamp

Information and Technology Management, Ruhr University of Bochum

Abstract

In the current discussion of the impact Web 2.0 may have on CSCW and Groupware research, Web 2.0 applications are often considered to be a substitute for Collaboration Support Systems. This paper argues that rather than replacing such systems, Web 2.0 mechanisms may complement them and help to overcome existing problems. The paper provides an analysis of differences and overlaps in purposes of applications and processes in these domains. This leads to the development of strategies for their combination. The paper describes real world implementations of these strategies and reflects upon how Web 2.0 may influence and even shape next generation Collaboration Support Systems.

1 Web 2.0 and Collaboration Support: Substitute or Major Upgrade?

Today's economy is vastly dependent on labor that is usually referred to by the notion of *knowledge work* (cf. Davenport 2005). Support for *knowledge workers* has to be accomplished on an individual and a group level, resulting in a transition from personal to group information management (Erickson 2006). However, despite influential and beneficiary outcomes of e.g. CSCW and Groupware research in this area, available *Collaboration Support Systems* (CSS) *still* lack important aspects needed for these tasks (McAfee 2006).

During the past years, we have experienced the rise of so called *Web 2.0* applications, in which a large number of users voluntarily engage in collaborative work. These applications show promising approaches towards overcoming lacks of traditional CSS. They provide an "architecture of participation" (O'Reilly 2005), which includes *simplicity of usage*, immediate *feedback* on UI and structural level as well as *valuing each user's contributions* (Grudin 2006). Web 2.0 orchestrates available technology in a way that encourages users to partici-

pate actively and helps to balance effort and benefit even in work-related settings[1]. The success of these applications – e.g. Wikis, Word Processors on the Web or Social Tagging systems – supports this point of view. *This immediately leads to the question whether Web 2.0 applications are the new generation of collaboration support systems.*

Replacing CSS by Web 2.0 applications is an appealing idea. In current discussions in research and practice, this is often seen as a solution to existing problems. Looking at the state of the art in CSS tells us why: Though CSS are commonly used in practice, they often lack adoption by users (Grudin 1988, Mark & Poltrock 2003). There are three main reasons for this: *First*, as Grudin (1988) puts it, in CSS there is a "disparity between those who do the work and those who get the benefit." *Second*, participation of users in collaboration is hindered by missing support for frictionless transitions between personal and group work. *Third*, integration of CSS into daily work and tool interoperability have to be improved. Web 2.0 applications seem to solve these problems, making them a tempting alternative to CSS.

Facing the question whether Web 2.0 can replace CSS, the discussion provided above only tells half of the story. This is because there is also a downside to Web 2.0 applications: features known from CSS are missing in some application types and known problems resurrect, resulting in poor support for e.g. awareness and communication[2], which are indispensable requirements for enterprise-grade applications. Tackling the question stated above needs an answer to the question whether current Web 2.0 applications are capable of covering all aspects of CSS. In this paper, we analyze these domains and conclude that this is not the case. We then argue that Web 2.0 applications and CSS can *complement* each other, resulting in synergies providing enhanced collaboration support. The resulting prototypes provide opportunities to evaluate the impact of Web 2.0 mechanisms to collaboration in existing CSS. However, as we have not conducted such an evaluation yet, the paper provides no data on that.

In what follows, we will provide a *conceptual* and *non-selective* distinction covering differences and dependencies between the domains of CSCW, Groupware and Web 2.0 applications. In section 3 we then take a pattern approach describing strategies suitable to meaningfully combine CSS and Web 2.0 applications for enhanced collaboration. We illustrate our considerations and their potential impact by three prototypical extensions to an existing CSS. Drawing from our experiences in designing and implementing them, in section 4 we discuss how the future of Web 2.0 enhanced collaboration support may look like and how resulting changes can be supported systematically. The paper concludes with our agenda for further research.

[1] See Millen et al. (2007) for an example on how social bookmarking services can be applied to improve search for information sources and social navigation in a corporate environment.

[2] It should be noted that some types of Web 2.0 applications such as Social Networking provide sufficient awareness support, whereas such support has to be improved in e.g. Social Tagging and Applications on the Web.

2 CSCW, Groupware and Web 2.0: Similarities, Differences and Synergies

Answering the question whether Web 2.0 applications may replace traditional CSS needs an understanding of differences and overlaps between terms and domains like Groupware, CSCW and Web 2.0. In the current discussion, there is often no such distinction, leading to different interpretations. However, it should be noted that there cannot be a *selective* distinction on these domains, and providing it is not what researchers in these fields should aim at. Instead, we should focus on the (original) *intentions* of these domains to find *conceptual* synergies between them. To accomplish this task, deriving differences and similarities from the usage and purposes of applications and processes of each domain is crucial.

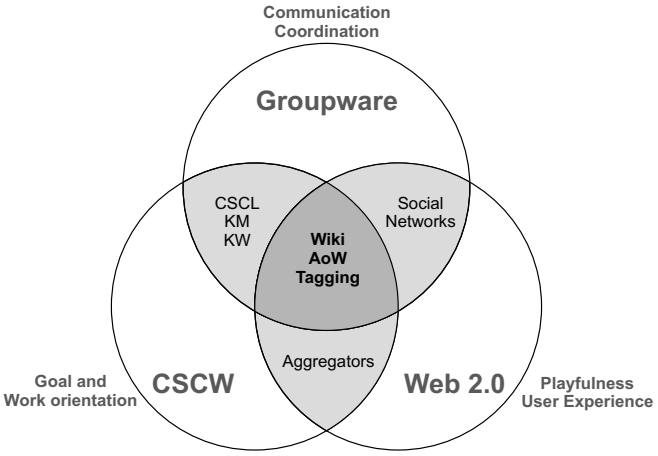

Figure 1: Differentiation of CSCW, Groupware and Web 2.0 applications

Figure 1 provides a conceptual distinction between the domains of CSCW, Groupware and Web 2.0 applications. The differentiation is based on three major characteristics present to a different extent in each domain: *goal and work orientation*, *communication and coordination* among peers in groups and *playfulness and user experience*. Considering CSCW, this domain is concerned with the support for specific, *work-related* tasks in groups (cf. Baecker 1992). Therefore, the driving factor for this domain is work or goal orientation. Taking document management or intranet portals as typical examples of this domain, communication is another but minor goal. Playfulness and user experience play a subordinate role. For Groupware, communication and coordination among peers has to be seen as the major characteristic. Groupware supports more *generic* tasks that are not necessarily associated with particular business goals or tasks (ibid). Applications in this domain may serve different goals or no goal at all, as can be seen by typical examples such as instant messaging or chat. There is also an emphasis on user experience, but aspects like playfulness and individual adoption are not as important as in Web 2.0 applications, which focus on individual *and*

group benefits. Examples like Wikis or Tagging communities show that in such applications there is no need for a particular goal. In such applications, communication support varies and thus cannot be considered a decisive factor for this domain.

Besides differences between these domains, there are also overlaps. By *CSS* we refer to the *overlap between CSCW and Groupware*. Synergy potential with Web 2.0 can therefore be found in the overlap between *all* domains. As can be seen in Figure 1, it is made up by *Wikis, Applications on the Web (AoW) and Tagging applications*. These applications all share a focus towards certain goals, a strong emphasis on communication and coordination as well as rich user experiences and playfulness in their usage. Therefore, learning from principles of these applications such as high user motivation provides strong potential for improving CSS. The other way round, CSS mechanisms provide good opportunities to improve Web 2.0 applications. Our work focuses on exploring such mutual benefits.

3 Integrating Web 2.0 and Collaboration Support

We have argued that challenges related to CSS and Web 2.0 applications can be overcome by finding synergies between them. Therefore, our work is concerned with implementing such synergies to demonstrate and later evaluate their impact. Here, we show three prototypes that add Web 2.0 functionality to KOLUMBUS 2 (*K2*, Prilla & Ritterskamp 2006), a CSS that has already been evaluated *on its own* in prior studies. Selecting K2 for the integration of Web 2.0 mechanisms in CSS offers the possibility to evaluate the effects of these functionalities in CSS. For the description, we will use a scheme consisting of the *problem* occurring in CSS, the *solution* approach made up by a combination of Web 2.0 and CSS, the *implementation* of a prototype in K2 and the *impact* potentially resulting from it.

3.1 Fostering Collaborative Work on the Web: Integrating Online Word Processors

Problem: Though collaboration support systems still heavily rely on secondary tools to e.g. produce the content that is managed by them[3], they lack proper application integration. This imposes an additional usage burden, as for creating and editing content a switch of applications is necessary. Instead of this extra effort, collaborative work should be fostered on the web, meaning that for e.g. content creation there should be convenient means to perform these tasks in collaboration support systems. This not only lowers the usage barrier but also contributes to user acceptance of the respective system.

Solution: Recently, Applications on the Web (AoW) such as Google Docs & Spreadsheets have become very popular. These applications enable users to fulfill several tasks known from desktop applications in a web browser – resulting in ubiquitous availability of content

[3] It should be noticed, however, that there are existing approaches allowing for e.g. content editing such as GROVE (Ellis et al. 1991), but these do not include the full set offered by secondary tools such as word processors.

creation and editing. The integration of AoW into CSS may provide a solution to the lack of application integration and the resulting application switches described above.

Implementation: The K2 Co-Writer was designed to support students in collaborative learning as well as professionals in science and business in collaborative writing. It is based on the integration of a web-based word processor into K2. To ensure proper support for content creation in CSS, the word processor was implemented to complement the existing system and adapted to its characteristics (see Figure 2).

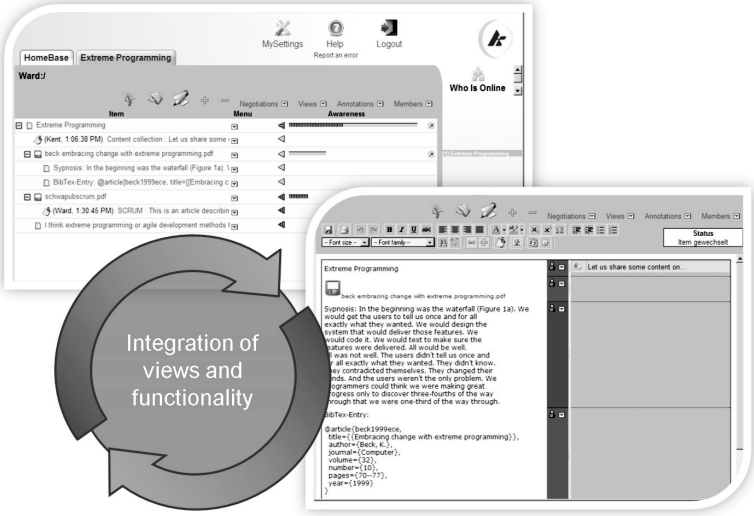

Figure 2: The Co-Writer extension

Whereas online word processors usually do not divide content into separate paragraphs that can be edited and owned by different users, the Co-Writer uses the fine-granular item structure of K2 and therefore enables multi-user support in asynchronous editing. We also integrated the contextualized communication and awareness features available in K2 to enhance the usability of the word processor. Furthermore, as shown in Figure 2, the word processor is implemented to be one of the different content views K2 provides. This way, users can decide whether to e.g. browse content or edit it.

Impact: The Co-Writer fulfils the requirements made up by the lack of content creation support in CSS in terms of collaborative writing (Prilla & Ritterskamp 2006). It is integrated into the system and provides a convenient way to edit and structure content, resulting in enhanced user experience. The other way round, this integration also diminishes AoW shortcomings like lacking awareness or communication support by using corresponding mechanisms *already present* in K2. The Co-Writer further illustrates how tool interoperability might foster knowledge work by reducing frictional losses resulting from frequent application switches. The effort for integrating such applications is moderate.

3.2 Enabling Users to control Scope and Impact of Content: Wiki-like Sharing

Problem: The creation of content in CSS is mainly limited to groups registered to these applications. While this makes sense in situations where content should be shared among a closed group, it prevents users outside these groups to contribute. Today, there is growing demands to involve parties external to a core group of collaborators in the creation and sharing of (selected) content. This holds true for diverse settings like distributed work in global software development (cf. Redmiles et al. 2007) and processes of open innovation (Chesbrough 2003). In any of these domains, the creation of content can benefit from contributions of (expert) users external to a group working in a certain application. With knowledge work increasingly depending on such participation in multi-project settings, corresponding functionality has to be provided by CSS.

Solution: Wiki-like content creation with its mixture of anonymous and known contributors provides a means to extend the scope of content production and sharing. The combination of publishing functionalities known from Wikis and sophisticated access control strategies known from the domain of CSS offers the potential of integrating formerly closed groups into a larger network of collaborators in a controlled manner.

Implementation: Learning from Wikis, we provided K2 with functionality to enable anonymous read and write access to certain parts of content. In the *"go public"* extension of the system, the creation of content can benefit from contributions of external users. Vice versa, results created internally may be well suited for public use. We therefore added functionality that allows users to negotiate public content and even enable anonymous access to it *within the application*. Whether content is to be published or not is always decided by the group producing the content: K2 supports such decisions by a negotiation mechanism that lets the assignment of access rights become subject to a group decision (Prilla & Ritterskamp 2006). The "go public" extension is combined with the Co-Writer for simplified content production.

Impact: Integrating extended Wiki-like publishing functionalities into collaboration support systems as described above enables users to easily enlarge the scope of visibility and contribution for certain content sections while other resources remain accessible only to a closed group. Multi-project knowledge work may benefit from functions that allow for project- and group-independent sharing and editing of content: by this means knowledge assets are no more bound to a single project but become available for others, too. Given that it is relevant to them, there is a fair chance that public content will be enhanced by the contributions of external users. Also, Wikis benefit from integrating sophisticated access control mechanisms, which is a prerequisite to their application in business settings.

3.3 Complementing existing Structures: Tagging Metadata for Collaboration Support

Problem: Finding content and making individual perspectives on it visible is still a problem

in collaboration support systems (Mathes 2004). Metadata is widely accepted to be the remedy for this problem. However, existing approaches using pre-defined metadata have shown poor user acceptance due to an imbalance of work and benefit (Grudin 1988), and usually result in trivial descriptions (Heath et al. 2005). Other approaches use formal semantics such as ontologies. While these mechanisms describe content properly, they impose an additional cognitive effort on users and may therefore not provide a solution to the problem (Golder & Huberman 2006, Grudin 2006).

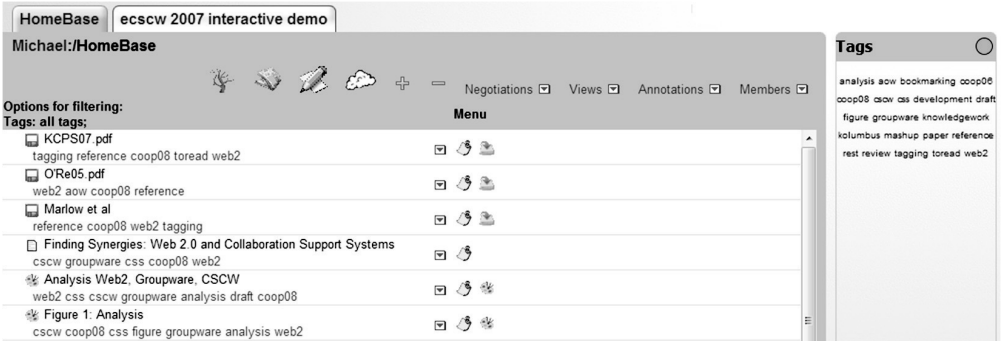

Figure 3: The Tagger plugin

Solution: Unrestricted content description mechanisms known from so called (Social) Tagging applications are now upon the most popular mechanisms on the internet. Their foremost strengths can be seen in serving different user purposes (Marlow et al. 2006) and in the low usage barrier they provide (Golder & Huberman 2006). Moreover, they provide meaningful content descriptions (ibid) and are capable of managing different content types (Prilla & Herrmann 2007), increasing the findability of information sources. Therefore, Tagging mechanisms are likely to solve the metadata problem in CSS.

Implementation: The K2 *Tagger* plugin aims at improvements for accessing, organizing and sharing different content types on a group and personal level by integrating Tagging into K2. It was designed to complement the existing content structure, meaning that well-known structures like folder hierarchies can be used in parallel to Tagging. Like the Co-Writer, the Tagger plugin provides an additional view on content as shown in Figure 3. Besides this view, the plugin includes a tag cloud (Rivadeneira et al. 2007) showing existing tags linked to content in the system. Furthermore, to foster the usage of tags, Tagging functionality is integrated in all system dialogues used for to create or edit content. Tagging here is not restricted to a single application: in K2, users are able to handle content from multiple applications *within a single structure*, allowing them to construct application independent content networks.

Impact: The integration of a Tagging mechanism may solve existing problems related to the lack of metadata in CSS. Overall, it enriches the means to structure and contextualize content. Content may now be browsed via folders *and* tags, resulting in a richer user experience. The ability to tag content from other applications also contributes to cross-application con-

tent integration. Tagging mechanisms provide a lightweight and unobtrusive means to support the creation of beneficial metadata both on a personal and group level: it therefore is a promising approach to intertwine a knowledge worker's personal information management with an organization's group information management processes. The other way round, Tagging mechanisms also benefit from functionality already present in collaboration support systems: not only can tags be contextualized by folder structures but also may mechanisms like awareness support and contextual communication enrich Tagging.

4 From Prototypes to systematic Integration

As can be seen from the descriptions in the previous section, collaboration support systems and Web 2.0 applications can complement each other in supporting tasks performed by knowledge workers. Reflecting our efforts in prototypically combining these applications has led us to both practical and theoretical considerations on the design of next generation CSS.

First, from the perspective of system development, the complex characteristics of knowledge work impose a huge variety of requirements on CSS and it is unlikely that the full potential of Web 2.0 enhanced collaboration support can be offered by a single *generic* application. As we have argued (Prilla & Ritterskamp 2008), we therefore suggest thinking of next generation systems as a flexible network of applications or services rather than of stand-alone environments. These networks may be tailored to the specific demands of *all* collaborations a knowledge worker is involved in, offering the potential to provide different selections of services that fit the demands from a multitude of projects at hand. CSS as Mashups have to be backed up by open and accepted architectural styles (e.g. REST, cf. Richardson & Ruby 2007) as well as by organizational support providing guidelines for building beneficial service networks and their deployment.

Second, organizational support for leveraging synergies between Web 2.0 applications and collaboration support systems can benefit from a systematic approach to providing problem-based patterns (Alexander 1977, Gamma et al. 1995, Herrmann et al. 2003). The description of the prototypes given in section 3 can serve as predecessors of such patterns. Though there is no empirical evidence that these solutions apply to more than one application setting (and can therefore be called *patterns*), other existing applications suggest their generalisability. For example, web-based word processors are becoming popular in Wiki applications, public content is available in some CSS as BSCW (Prinz et al. 2008) and Tagging mechanisms have been shown to provide improvements on an enterprise level (Farrell et al. 2007, Millen et al. 2007). Future research should be aimed at both scrutinizing the pattern candidates presented in this paper and extending the pattern collection describing beneficiary synergy potentials of Web 2.0 applications and collaboration support systems.

5 Outlook

There remain further research questions for Web 2.0 enhanced CSS. First, although a pattern approach may serve well in describing *when* and *how* a specific solution can be applied, it does not necessarily answer the question *why* this solution will work. We have argued that the architecture of participation Web 2.0 applications promote may reduce the gap between effort and benefit that oftentimes hinders the adoption of common CSS. Future research should aim to verify this assumption and identify further characteristics contributing to the successful organizational deployment of CSS. The prototypes described in this paper provide the basis for such an evaluation. Second, the idea of CSS as Mashups raises questions that transcend the level of architectural styles. For instance, with vendors offering specialized services whose integration into customer-specific networks may be subject to charges, there is also a promising economic perspective requiring suitable service level agreements, payment models and security concepts. Finally, the possibilities and conditions of fostering cooperation in corporate networks and open innovation processes with Web 2.0 enhanced CSS are topics worth considering.

References

Alexander, C., Ishikawa, S., Silverstein, M., Jacobson, M., Fiksdahl-King, I., & Angel, S. (1977). *A pattern language: towns, buildings, construction.* Oxford: Oxford University Press.

Baecker, R. (1992). *Readings in Groupware and Computer-Supported Cooperative Work: Assisting Human-Human Collaboration* (Preface). Morgan Kaufmann.

Chesbrough, H. W. (2003). *Open innovation*. Boston: Harvard Business School Press.

Davenport, T. H. (2005). *Thinking for a living.* Boston: Harvard Business School Press.

Ellis, C., Simon, J., & Rein, G. (1991).Groupware: some issues and experiences. *Communications of the ACM 34*, 39-58.

Erickson, T. (2006). From PIM to GIM: personal information management in group contexts. *Communications of the ACM 49*, 74-75.

Farrell, S., Lau, T., Wilcox, E., Nusser, S., & Muller, M. (2007). Socially augmenting employee profiles with people-tagging. In: Chen, C., Jacob, R., Balakrishnan, R. (eds.): *Proc. UIST 2007*, pp. 91-100. New York, NY: ACM.

Gamma, E., Helm, R., Johnson, R., & Vlissides, J. (1995). *Design patterns: elements of reusable object-oriented software*. Boston: Addison-Wesley Longman Publishing Co., Inc.

Golder, S. & Huberman, B. A. (2006). Usage patterns of collaborative tagging systems. *Journal of Information Science 32*(2), 198-208.

Grudin, J. (2006). Enterprise Knowledge Management and Emerging Technologies. In: Sprague, R. H. (ed.): *Proc. of HICSS '06*, pp. 57.1. Washington, DC: IEEE.

Grudin, J. (1988). Why CSCW applications fail: Problems in the Design and Evaluation of Organizational Interfaces. In: *Proc. of CSCW 1988*, pp. 85-93. New York, NY: ACM.

Heath, B. P., McArthur, D. J., McClelland, M. K., & Vetter, R.J. (2005). Metadata lessons from the iLumina digital library. *Communications of the ACM 48* (7), 68-74.

Herrmann, T., Hoffmann, M., Jahnke, I., Kienle, A., Kunau, G., Loser, K., & Menold, N. (2003). Concepts for usable patterns of groupware applications. In: Pendergast, M., Schmidt, K., Tremaine, M. & Simone, C. (eds.): *Proc. of GROUP 2003*, pp. 349-358. New York, NY: ACM.

Mark, G. & Poltrock, S. (2003). Shaping Technology Across Social Worlds: Groupware Adoption in a Distributed Organization. In: Pendergast, M., Schmidt, K., Tremaine, M. & Simone, C. (eds.): *Proc. of GROUP 2003*, pp. 284-293. New York, NY: ACM.

Marlow, C., Naaman, M., Boyd, D., & Davis, M. (2006). HT06, tagging paper, taxonomy, Flickr, academic article, to read. In: Wiil, U. K., Nürnberg, P. J., Rubart, J. (eds.): *Proc. of the seventeenth conference on Hypertext and hypermedia*, pp. 31-40. New York, NY: ACM.

Mathes, A. (2004). *Folksonomies – Cooperative Classification and Communication Through Shared Media.* Urbana-Champaign: Graduate School of Library and Information Science, University of Illinois.

McAfee, A. (2006). Enterprise 2.0: The Dawn of Emergent Collaboration. *MIT Sloan Management Review 47*, 21-28.

Millen, D., Yang, M. Whittaker, S., & Feinberg, J. (2007). Social bookmarking and exploratory search. In: Bannon, L., Wagner, I., Gutwin, C., Harper, R., Schmidt, K. (eds.): *Proc. of ECSCW'07*, pp. 21-40. London: Springer.

O'Reilly, T. (2005). *What is Web 2.0. Design Patterns and Business Models for the Next Generation of Software*. http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html.

Prilla, M. & Herrmann, T. (2007). Semantically Integrating Heterogeneous Content: Applying Social Tagging as a Knowledge Management Tool for Process Model Development and Usage. In: Tochtermann, K. & Maurer, H. (eds.): *Proc. of the 7th International Conference on Knowledge Management*, pp. 16-24. Graz: JUCS.

Prilla, M. & Ritterskamp, C (2006). Collaboration Support by Co-Ownership of Documents. In: Hassanaly, P., Herrmann, T., Kunau, G. & Zacklad, M. (eds.): *Proc. of COOP 2006*, pp. 255-269. Amsterdam: IOS.

Prilla, M. & Ritterskamp, C. (2008). Finding Synergies: Web 2.0 and Collaboration Support Systems. In: Hassanaly, P., Ramrajsingh, A., Randall, D., Salembier, P. & Tixier, M. (eds.): *Proc. of COOP '08*, pp 35-41. Aix-en-Provence: Institut d'Etudes Politiques.

Prinz, W., Hinrichs, E., & Kireyev, I. (2008). Anticipative Awareness in a Groupware System. In: Hassanaly, P., Ramrajsingh, A., Randall, D., Salembier, P. & Tixier, M. (eds.): *Proc. of COOP '08*, pp. 5-15. Aix-en-Provence: Institut d'Etudes Politiques.

Redmiles, D., van der Hoek, A., Al-Ani, B., Hildenbrand, T., Quirk, S., Sarma, A., Filho, R. S. S., de Souza, C., & Trainer, E. (2007). Continuous Coordination – A New Paradigm to Support Globally Distributed Software Development Projects. *Wirtschaftsinformatik 49* (Sonderheft), 28-38.

Richardson, L. & Ruby, S. (2007). *RESTful Web Services.* Sebastopol: O'Reilly Media.

Rivadeneira, A. W., Gruen, D. M., Muller, M. J., & Millen, D. R. (2007). Getting our head in the clouds: toward evaluation studies of tagclouds. In: Rosson, M. B. & Gilmore, D. (eds.): *Proc. of CHI 2007*, pp. 995-998. New York, NY: ACM.