Position paper: Adapting Web 2.0 to Corporate Reality

Kristian Tørning
The Danfoss Group – Danfoss Business System
Nordborgvej 81– DK 6430 Nordborg – Denmark
toerning@danfoss.com
(+45) 51 51 29 32

ABSTRACT
This position paper argues that we need new design methods for producing enterprise CSCW systems. It is speculated that we might learn from social structures and interaction as seen in Web 2.0 communities and employ them as lever for enhancing enterprise systems adoption and content contribution rates.

INTRODUCTION
When designing CSCW systems for globally distributed organizations there is a need to consider the emotional value proposition that the systems offers the Knowledge Workers that are to use it. In particular there is a need to design for motivation. On the Internet Social Software caters as much to emotions as to logic, but most current day software design methods are mostly concerned with the logic in the applications, rather than on the emotional impact of the application or even its entertainment value.

USER CENTERED DESIGN
The most popular methods for user experience design are currently based on the User Centered Design paradigm, thus placing the end-user at the center of all efforts in the system design. User Centered Design is the de facto industry standard i.e. Oracle, SAP, SUN IBM and Microsoft (see reference 1 in reference section).

User Centered Design (UCD) can be defined as the: "…active involvement of users for a clear understanding of user and task requirements, iterative design and evaluation, and a multi-disciplinary approach" (Mao et al, 2001). With UCD computer systems are built to satisfy end-users genuine needs for solving specific logic tasks in an environment, thus taking into account the context that the end users have to act in (Raskin, 2000). UCD typically addresses the specific tasks that the end-user is *supposed to be solving*. Preferably these tasks are uncovered or validated by observing employees conducting their work. Most UCD methods that we might employ when designing enterprise software are more focused on human-to-computer interaction in the sense that they aspire to achieve the highest level of usefulness and usability in regards to solving tasks. A central design criterion is normally not modeling a superior emotional experience – or just an entertaining one.

DESIGNING FOR KNOWLEDGE WORKERS
Possibly the main challenge when designing for Knowledge Workers is that they have a lot of freedom in what they are actually doing and how they are doing it. Most Blue Collared Workers will to a larger extent have to operate the interfaces (systems) they are given and they might also have a clearer set of tasks to solve. The cash register or the mobile device used for registering crates in a warehouse setting would often be rather fixed elements in their workflow. In comparison Knowledge Workers have more freedom of choice; they might choose not to adopt a CSCW system provided by their organization, if they don't feel there is something in it for them (Bansler and Havn, 2002). For instance they might cover their document sharing needs by sending emails rather than adopting the SharePoint workspaces, effectively leaving all the knowledge and documentation sitting on a mail server non-accessible to anyone else in the organization. This is unique and I would argue that it requires additional design considerations when designing for instance systems for knowledge sharing.

Motivation
Knowledge Workers can often be suffering from 'Time Famine'. Time famine is a term coined by Leslie Perlow (Amabile et al, 2002) referring to the fact that Knowledge Workers often have more than a full plate, when it comes to activities that they might be expected to engage in. The same people that organizations expect to: invent the new products, services and the organizations of tomorrow are under enormous time pressure. This is naturally a problem when designing systems to support innovation and ideation since research indicates that employees suffering from time famine would be less likely to get fresh ideas (Amabile, 2002). Furthermore the ever growing demands of enhanced efficiency have a negative effect on intrinsic motivation (Amabile, 1998). This poses a problem since intrinsic motivation is often what drives people to perform, for instance staying up late working on a concept or project.

Bluntly stated, it would seem overly optimistic to expect Knowledge Workers to readily supply: ideas, innovations concepts or specialized domain knowledge of business processes, if the corporation they are in do not all ready have a well established corporate culture of sharing information and ideas (Cynthia and Harrington, 2001).

Attention and intrinsic motivation are key factors in designing systems for Knowledge Workers, if there are many things they could be doing, why would they go to a CSCW system and start documenting business processes or possibly filing their ideas (for others to steal)? The hardest part of designing a knowledge management system for knowledge sharing and collaboration is to actually have the system adopted by the end-users. The users will have to like the system so much, that they will prioritize spending time on supplying content such as: ideas, suggestions and process documentation. The main challenge becomes to address the intrinsic motivation at *design time*, so that the attention of the Knowledge Workers can be captured and maintained in a sustainable way. How do we solve that task? How do we design enterprise software for motivation and attention?

SOCIAL SOFTWARE HAS MOTIVATED USERS

I would argue that today the CSCW designer's toolbox is mostly empty, when it comes to directly addressing motivation, entertainment and attention in an explicit and formal manner. I believe we lack design methods that would enable us to address such aspects of design in a deliberate fashion. But in comparing enterprise CSCW systems to social software (Web 2.0), the end-users of the latter are spending huge amounts of time contributing and spending load of attention on other users content. In websites like: Myspace.com, Facebook.com and YouTube.com intrinsic motivation is soaring and these websites have engaged end-users that are contributing with plenty of content. This type of 'social'-software seemingly caters to some basic human need of: sharing, interacting, communicating and socializing with others. Although the mentioned applications are built for entertainment purposes more than collaboration as such, it would seem that they somehow make the users contribute with enormous amount of content. In general Web 2.0 applications address the end-user *intrinsic motivation* in ways that are currently difficult to describe and even more difficult to prescribe. What would a motivational design framework for instance look like for CSCW systems? How could we somehow tap into some of that power when designing software for the work place?

POSITION

Current system design methods don't explicitly recommend the possible use of informal communication as a lever for system adoption but we can observe Social Web 2.0 systems with informal communication that nearly turns some of its users into system-addicts. In theory it should be possible to tap into the dedication found in social Web 2.0 software when we are designing internal CSCW systems for Knowledge Workers. When we are designing systems that are to facilitate collaboration and content sharing there is a need to better understand the human-to-human computer mediated relationship, rather than the human-to-technology relationship for task solving that many modern design methods mostly emphasize. We need to be able to give users an opportunity to experience each other via technology rather than only focusing on how people experience and interact with the technology itself. Naturally usability and Human Computer Interaction cannot be ignored, but it might not be the main challenge or most important design criterion, when designing systems, that are to engage multi-disciplinary Knowledge Workers globally.

REFERENCES

1. http://ui.oracle.com/, https://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uuid/ba649 466-0d01-0010-488a-d95cac6e0631 http://www.sun.com/usability/index.html http://ibm.com/services/us/igs/pdf/ucd-at-ibm-consulting.pdf http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnanchor/html/anch_uidesigndev.asp
2. Amabile, T. M., Hadley C, N., Kramer S. J. (2002): 'Creativity Under the Gun', HBR On Point – from the Harvard Business Review A collection of Articles "Best of HBR on Rescuing Your Company's Creativity", Product Number 1563, pp. 3-12
3. Amabile, T. M., (1998): 'How to Kill Creativity', HBR On Point – from the Harvard Business Review A collection of Articles "Best of HBR on Rescuing Your Company's Creativity", Product Number 3499, pp. 76-87
4. Bansler, J. P., & Havn E. C., (2002): 'Knowledge Sharing in Heterogeneous Groups: An
5. Empirical Study of IT-Support for Sharing Better Practices', Third European Conference on Organizational Knowledge, Learning and Capabilities, 2002
6. Cynthia P. R., and Harrington S. J., (2001): 'Sharing: Knowledge Through Intranets: A study of
7. Organizational Culture and Intranet Implementation', IEEE Transactions on Professional Communication, vol. 44. No. 1, March 2001, pp. 37-51
8. Examining the information architecture in a law enforcement agency', Journal of the American Society for Information Science and Technology, 53(10), 846-854 (2002)
9. Johansen, R. (1988): Groupware. Computer Support for Business Teams, The Free Press, New York and London.
10. Mao, J. y., Smith P. W., Vredenburg K., Carey T., (2001): 'User-Centered Design Methods in
11. Practice: A Survey of the State-of-the-Art', IBM Press (2001)
12. Raskin J. (2000): Designing the Humane Interface, Addison Wesley, New York