Co-activity manager: integrating activity-based collaboration into the desktop interface

Peer-reviewed author version

Houben, Steven; VERMEULEN, Jo; LUYTEN, Kris & CONINX, Karin (2012)
Co-activity manager: integrating activity-based collaboration into the desktop interface. In: Tortora, Genny; Levialdi, Stefano; Tucci, Maurizio (Ed.). Proceedings of the International Working Conference on Advanced Visual Interfaces (AVI '12), p. 398-401.

DOI: 10.1145/2254556.2254633
Handle: http://hdl.handle.net/1942/13889

co-Activity Manager: Integrating Activity-Based Collaboration into the Desktop Interface

Steven Houben[1] Jo Vermeulen[2] Kris Luyten[2] Karin Coninx[2]

[1]The Pervasive Interaction Technology Laboratory – IT University of Copenhagen
[2]Expertise Centre for Digital Media – Hasselt University – tUL – IBBT
[1]shou@itu.dk, [2]{jo.vermeulen, kris.luyten, karin.coninx}@uhasselt.be

ABSTRACT

Activity-Based Computing (ABC) has been proposed as an organisational structure for *local* desktop management and knowledge work. Knowledge work, however, typically occurs in partially overlapping subgroups and involves the use of multiple devices. We introduce co-Activity Manager, an ABC approach that (i) supports activity sharing for multiple collaborative contexts, (ii) includes collaborative tools into the activity abstraction and (iii) supports multiple devices by seamlessly integrated cloud support for documents and activity storage. Our 14 day field deployment in a multidisciplinary software development team showed that activity sharing is used as a starting point for long-term collaboration while integrated communication tools and cloud support are used extensively during the collaborative activities. The study also showed that activities are used in different ways ranging from project descriptions to to-do lists, thereby confirming that a document-driven activity roaming model seems to be a good match for collaborative knowledge work

Keywords

Activity-Based Computing, Collaborative Work, Activity Sharing, Instant Messenger, Activity Cloud

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces, graphical user interfaces, windowing system

1. INTRODUCTION

Knowledge work typically consist of both *individual and highly collaborative work*. Figure 1 shows a multidisciplinary team of knowledge workers who collaborate on different projects with different colleagues in partially overlapping subgroups (*a*, *b* and *c*). Team members continuously collaborate on several shared parallel activities that require some form of synchronization (e.g. sharing files and being aware of each others updates on those files) between individual work. Many knowledge workers thus might want to tailor their part of the work according to their personal preferences. In order to collaborate on these projects and share individual work, knowledge workers use a number of different tools (e.g. email, instant messengers), which are usually chosen based upon personal preference and in agreement with collaborators. Studies however have shown that interruptions caused by these tools often result in project fragmentations. Additionally, people nowadays also incorporate multiple devices into their lives, which requires them to deal with the burden of managing information and activities across all these devices [3].

Several approaches aim to address the problems of knowledge workers in the digital age by integrating activity management into the desktop interface (e.g. [4, 8, 9]). Project Colletta [7], Giornata [10] and the ABC system [1] closely integrate with the operating system by using a virtual desktop like system as structuring mechanism for activities. The latter two also consider communication and collaboration. Giornata [10] provides a contextually populated *contact pallette* that can be used to share files via email and also serves as a visual cue on the amounts of unread emails. In the ABC system, file sharing and real-time collaboration is supported through a pervasive framework that was designed for hospital environments [2]. Finally, Activity Explorer [6] successfully introduced an activity sharing system but limits its approach to predefined objects that are confined inside the application. In summary, no existing approach currently focusses on sharing the entire *activity* desktop workspace for seamless collaboration and availability on multiple devices nor on integrating synchronous communication tools such as instant messengers into the activity abstraction to reduce project fragmentation.

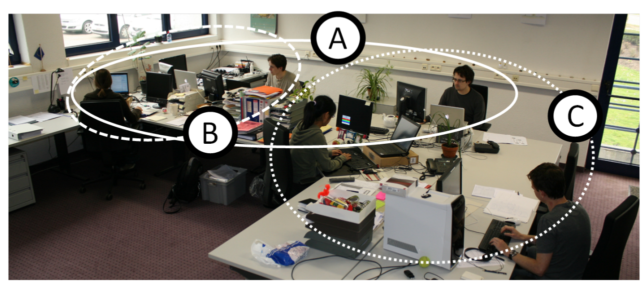

Figure 1: A multidisciplinary software development team. The groups (*a*, *b*, *c*) symbolize some of the collaborative relations between members.

Since Activity-Based Computing (ABC) relies on the activity concept as an information abstraction, knowledge workers should be able to seamlessly share entire activities with other team members on top of the collaborative functionality on a document level. Ac-

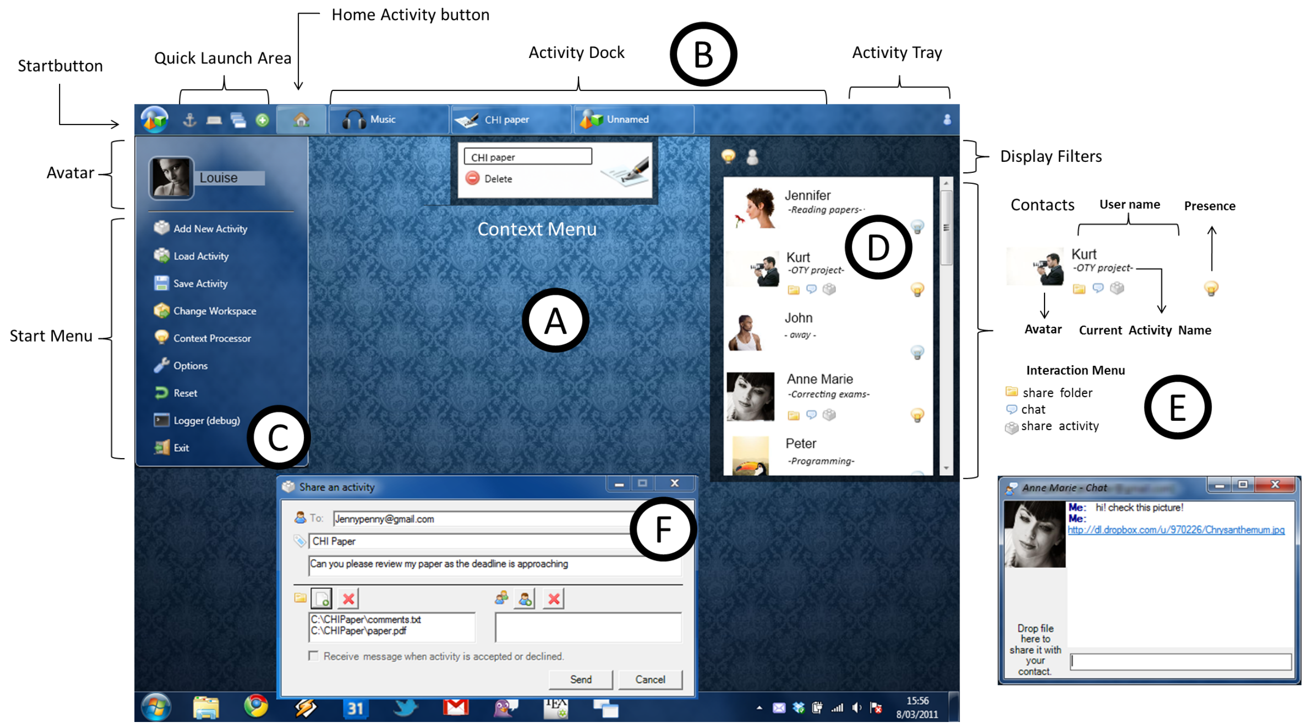

Figure 2: The interface of co-Activity Manager consists of (A) a per-activity workspace, (B) an activity task bar to visualize activities to the user, (C) an activity start menu to manage activities and applications, and (D) a collaboration manager to interact and share with contacts. Each contact is visualized with an avatar, name and status field. The interaction menu (E) can be used to share a folder, chat or share an activity (F).

tivities should therefore be available in and deployable from the cloud [5]. In addition, this would also make it easier for users to access their activities and information from multiple devices, thereby addressing the problem of multi-device fragmentation [3]. Finally, communication tools should closely integrate into the ABC paradigm to avoid forcing users to deal manually with unrelated communication interruptions while performing collaborative activities.

In this note, we introduce an Activity-Based Computing (ABC) approach that (i) supports activity sharing for multiple collaborative contexts by allowing users to share and deploy an activity-based desktop workspace; (ii) includes collaborative tools (including file sharing, messaging and collaboration) into the activity abstraction; and (iii) provides a lightweight cloud mechanism that allows users to save activities on one instance of cAM and open them in another instance on a different computer.

2. CO-ACTIVITY MANAGER

co-Activity Manager (cAM) (Figure 2) is an activity-based desktop manager that augments the Microsoft Windows 7 *window- and task-manager*. cAM deals with workflow fragmentation and communication interruptions by allowing the user to (re-)organize documents, applications and files as well as structuring communication and collaboration with other participants in shared activities. The purpose of cAM is to minimize *out-of-context* interrupts by restructuring the desktop and *filter* communication based on the current activity. For each activity, a separate virtual desktop is created that confines the workflow of that activity. The activity workspace (Figure 2 A) allows users to pile relevant files and documents which are then automatically related to the activity. An activity taskbar (Figure 2 B) is used to manage and work on activities. Users can create, save, edit and manage local activities as they see fit (Figure 2 C).

2.1 Communication and Collaboration

cAM includes a collaboration manager (Figure 2 D) to facilitate and structure activity-based communication and collaboration. The manager supports standard *chat* messages. The chat window is equipped with an automatic sharing system, that allows users to drag and drop documents they want to share on top of the chat windows. These files are then automatically uploaded to the cloud storage and shared via a web link. The user can also define a *shared folder* for each contact per activity. All these folders are stored in the cloud and can be used as a persistent sharing mechanism or to share large amounts of data. Users can also *share activities*, which will be discussed in the next section.

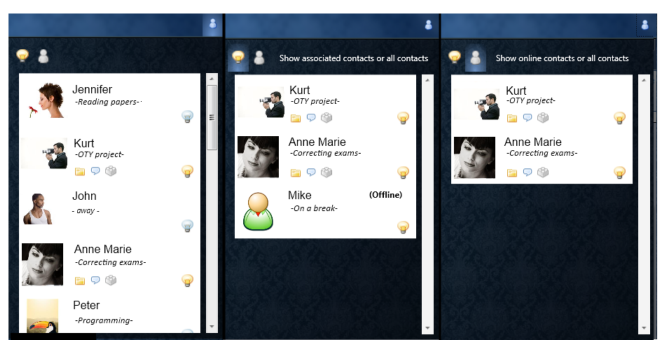

Figure 3: A contact can be added to or removed from the current activity by clicking the light bulb icon. It is only possible to interact with contacts who have added you to their activity as well (as indicated by the interaction menu below the contact's name).

Because the collaboration manager is integrated in the activity manager, the list of contacts can also be repopulated per activity. This means that a user can select the contacts that are relevant for each specific activity. By adding these contacts to the collaboration list of her current activity, she defines a group of contacts who are allowed to communicate with her and whose status is shared in the context of that activity. The system also distributes the name of the ongoing activity to all related contacts. When the user switches between activities, the collaboration list is updated and the contacts will see her appear as offline or online depending on whether they are added to her loaded activity or not. If the user is online, the name of the activity she is currently engaged in will be shown to others. This activity-based communication filtering allows users to control not only their workflow but also their communication flow. By allowing the user to control who can interrupt her, she can also better control her entire workflow, thereby decreasing task and workflow fragmentation. In order to add or remove a contact, the user simply has to click the light bulb button that is located on the interaction menu (Figure 2 E) of each contact. The collaboration manager also provides association and online/offline filters (Figure 3) to customize the list according to personal preference for each activity. These filters remove or show offline/online or associated/not associated contacts. In order to distribute these activity-based presence changes to all contacts, we use a custom-built XMPP extension.

2.2 Activity Sharing

The user can not only structure her own ad-hoc workflow in activities but can also share these activities with her collaborators. To accomplish this, she launches the collaboration manager (Figure 2 D), clicks the "share activity" button in the interaction menu of the contact she wants to share the activity with (Figure 2 E), and enters a description for that activity in the "share an activity" window (Figure 2 F). These descriptions can include a name, a description of the motivation for and goals of the activity, a set of resources (e.g., local documents, folders, applications), relevant contacts and the activity history that lists all updates and changes that have been made to the activity. After the related resources are stored in the cloud, the system sends the activity to the selected contact(s). The receiving user is notified that a new activity is made available by another user and can then choose to deploy it as a local desktop activity or queue it for later usage.

2.3 Loading and Saving Activities in the Cloud

Next to sharing activities with collaborators, users can also save their own activities in the cloud and load them later on other devices through the activity start menu (Figure 2 B). This means that files, documents and activities can be stored online and accessed seamlessly through the desktop interface. Our system creates an Open Activity XML (OAXML) description of an activity and all of its related resources and saves this description and all related data into the cloud. The user can then import the same activity on another device that is equipped with software that supports our OAXML format. In the current implementation we used the Dropbox API [1] to provide cloud storage, but since the activity management is done on the client side, any cloud platform can be used.

3. FEASIBILITY STUDY

To assess the feasibility and usefulness of cAM in a real work setting, we deployed co-Activity Manager in a multidisciplinary software development team (seen in Figure 1) that specializes in developing interactive setups for cultural heritage sites such as museums. The team consists of five people in total: three software engineers (P1, P2 and P3), a graphic designer (P4) and an historian (P5) who were instructed to use the system for their day-to-day work over a period of two weeks. During this period, we observed and interviewed participants periodically to detect potential problems that would hinder further participation in the study. At the end of the deployment, we conducted semi-structured interviews to discuss participants' experiences with cAM and elicit detailed feedback about the activity-based collaboration and communication features.

3.1 Activities as organisational structure

Four out of five participants indicated that they really liked using activities. In contrast with his colleagues, however, P3 was initially quite reluctant to change his way of working. P1 felt that activities were very useful because they allowed him to structure his workflow based on the parallel ongoing projects. He also felt that using activities helped him focus better on his work. The historian (P5) used activities in a rather unexpected way. At the beginning of each workday, she created a set of activities which mapped directly on tasks she was planning to do that day. P5 was essentially using activities to create her daily *to do list*. Although we did not anticipate this kind of usage, this approach worked very well for her and also demonstrates the flexibility and benefits of using the activity concept as an organisational unit.

3.2 Activity sharing to start collaboration

Activity sharing did not occur very frequently but was rather used as a *starting point* for a new project or task. For example, because P1 received new information – documents and images related to a specific project that were also important for P2, but also a list of all related contacts – he created a new activity with all this information and shared it with P2, who then deployed it as a local activity. Both of them liked that instead of having to copy documents over manually, they could simply *share an entire context* which would then automatically be deployed on a new desktop. We also noticed that after the activity was shared, P2 reorganized her desktop to match her own work style. During the interview, she explained that because all the relevant information was already on the desktop, reorganizing it was easy. P3 also noted that handing over activities asynchronously could be useful for transferring (partial) workflows between users.

3.3 The per-activity collaboration list

During the interview it also became clear that participants appreciated the integrated collaboration features, such as folder sharing and per-activity contact lists. Participants felt that the per-activity contact list was a useful feature, especially for people that do a lot of tasks or projects at the same time. More specifically, P5 mentioned that she found this way of working very useful as it allows her to filter out irrelevant contacts so that all information and relevant persons are bundled in a single desktop view. Not all participants were convinced by the chat functionality, though. P2 argued that she did not use IM very often as she was physically co-located in the same room as her colleagues. She did, however, use the per-activity contact list as a starting point for file and activity sharing. P1 confirmed that the IM functionality was a *second communication channel* as a lot of the discussion was done face to face. He also requested to add functionality that would allow certain contacts to be automatically added to all activities (which we call "Chuck Norris contacts")[2] as he wanted to be available to his wife at all times.

[1] https://www.dropbox.com/developers

[2] Because Chuck Norris is relevant to *all* your activities.

A number of issues arose from the mixture of private and work-related communication. Participants' activity-based presence was sent to all their Google Talk contacts rather than to the ones that also used cAM, which raised some privacy and confidentiality concerns. P2 argued:

> *"because the name of the ongoing activity is shared with all contacts, there is a potential danger of leaking confidential information as some people might use the name of unannounced projects as name for their activities".*

P3 also renamed one of his activities to be work-related, even though it in fact only contained personal content, as he did not want his colleagues to be informed of this. Finally, the interviews also showed that the per-activity contact list would at times cause inconvenient situations. Participants sometimes switched activities during a conversation which would result in muting the other contact (if they did not happen to be part of that other activity). The muted contact would then have no way of leaving a message or response. P2 argued that asynchronous messaging therefore should be integrated into the system (as also proposed earlier by Voida et al. [10]), but P1 disagreed with this as he argued that email could be easily launched as part of the ongoing activity.

3.4 Multi-device support through the cloud

Activity-centric cloud support was perceived as *very useful*. Most participants had two to four different devices, so they all liked the idea of storing activities in the cloud and deploying them on different machines. P2 liked the fact that she could access her documents from different places at the same time. P3 noted that he could now save his activity on his main machine and resume it on another device. Finally, P1 did not only like cloud support because it could be used to distribute activities over multiple devices, but also because the cloud storage mechanism allowed him to backup contextually meaningful structures rather than just a set of files.

4. DISCUSSION

Deploying cAM provided some promising results and allowed us to identify several opportunities and shortcomings of our current approach. In what follows, we give an overview of lessons learned and outline possible directions for future research.

The study indicated that the inclusion of an integrated cloud system is very useful. As most participants already tend to rely on remote servers or cloud storage to save their documents, they generally *liked being able to store activities in the cloud*. It allows them to move beyond storing documents and save meaningful structures in the cloud. It also opened up *opportunities for resuming activities on multiple instances of cAM* or even use the activity structure as a back up mechanism.

Participants used the *activity-centric collaboration* tool as intended: they created a customized contact list that functioned as a starting point for activity sharing and activity-based interaction. Our focus with cAM is currently on real-time communication (instant messaging), but during the field study several participants suggested that an *asynchronous* activity-based messaging system (e.g. integrated email) would be a useful addition, thereby confirming earlier findings by Voida et al. [10]. In summary, activity-based collaboration occurred in two phases: an instantiation phase in which the activity workspace tool was use to *configure* the collaboration; and the collaboration phase in which the per-activity collaboration tools were used to actually *consume* the collaboration inside the shared activity.

A number of issues arose regarding *privacy* and *confidentiality*. Further investigation exposed a deeper problem that is rooted in the balance between organisational policies and personal preferences. The company for which the team that performed the evaluation works, allows its employees to use instant messengers *both* for personal and work-related purposes. This caused some concerns on how work-related information might be shared with private contacts, and private information might become shared with work colleagues. One solution to this problem would be to provide a mechanism *to distribute activity awareness only to relevant contacts* (thereby creating a closed activity sharing group) and allow users to *control themselves with whom they share their activity presence*. This mechanism could also be made available at an organisational level to minimize the potential danger of leaking confidential information. There is thus an important balance between freedom of use and automatic awareness.

Participants appreciated the focus cAM puts on sharing the *document flow* since this provides them with the flexibility to continue working with tools they are familiar with. Personalizing the shared desktop after accepting the activity was also found to be considerably easier than downloading all corresponding files, setting up the correct instant messenger contact list and finding the right documents in the task bar. Our study showed that participants use activities in very different ways including as to do lists, detailed project descriptions or simply as a single desktop in which everything was organized. Additionally, participants also created both anonymous activities for volatile use as well as activities with a detailed description for long-term use. In conclusion, our approach to Activity-Based Computing allows users to structure collaboration and share activities without losing the flexibility to tailor work according to their own preferences.The *document-driven activity roaming model* seems to be a good match for collaborative knowledge work and a valuable structuring mechanism for desktop workspace sharing.

5. REFERENCES

[1] J. Bardram, J. Bunde-Pedersen, and M. Soegaard. Support for activity-based computing in a personal computing operating system. In *Proc. of CHI '06*, pages 211–220. ACM.

[2] J. E. Bardram. Activity-based computing for medical work in hospitals. *ACM Trans. Comput.-Hum. Interact.*, 16:10:1–10:36, June 2009.

[3] D. Dearman and J. S. Pierce. It's on my other computer!: computing with multiple devices. In *Proc. on Human factors '08*, CHI '08, pages 767–776, New York, NY, USA. ACM.

[4] V. Kaptelinin. Umea: translating interaction histories into project contexts. In *Proc. of CHI '03*, pages 353–360. ACM.

[5] V. Kaptelinin and M. Czerwinski, editors. *Beyond the Desktop Metaphor: Designing Integrated Digital Work Environments*, volume 1 of *MIT Press Books*. The MIT Press, 2007.

[6] M. J. Muller, W. Geyer, B. Brownholtz, E. Wilcox, and D. R. Millen. One-hundred days in an activity-centric collaboration environment based on shared objects. In *Proc. of CHI '04*, pages 375–382. ACM.

[7] G. Oleksik, M. L. Wilson, C. Tashman, E. Mendes Rodrigues, G. Kazai, G. Smyth, N. Milic-Frayling, and R. Jones. Lightweight tagging expands information and activity management practices. In *Proc. of CHI '09*, pages 279–288, New York, NY, USA. ACM.

[8] T. Rattenbury and J. Canny. Caad: an automatic task support system. In *Proc. of CHI '07*, pages 687–696. ACM.

[9] G. Smith, P. Baudisch, G. Robertson, M. Czerwinski, B. Meyers, D. Robbins, and D. Andrews. Groupbar: The taskbar evolved. In *Proc. of OZCHI 2003*, pages 34–43.

[10] S. Voida, E. D. Mynatt, and W. K. Edwards. Re-framing the desktop interface around the activities of knowledge work. In *Proc. of UIST '08*, pages 211–220. ACM.