

Activity-Based Computing

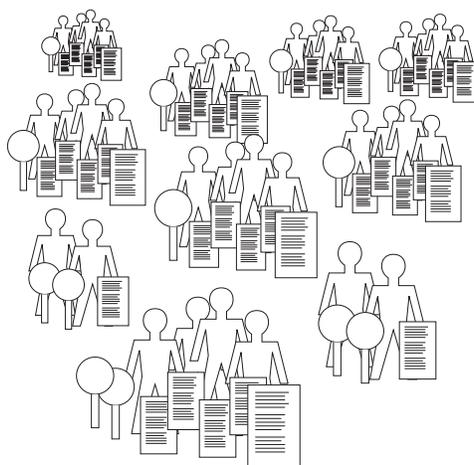
John Canny and Danyel Fisher

Computer Science Division, UC Berkeley

When working in shared physical spaces, individuals develop a rich sense of awareness. They gather and share information freely; they gain a sense of what others know which allows them to ask the right person for help; they gain a sense of what others are working on which allows them to provide pertinent help; and they come to understand other's goals which gives them a richer sense of purpose. This richness seems to be missing in electronic contexts. We propose a notion of high-level context that is versatile and productive.

Our approach is called "Activity-Based Computing" or ABC. ABC has two key components: (i) the interface and conceptual model is based on high-level activities rather than actions, and (ii) the use of tacit data to infer a rich set of relationships among people, documents and tools. ABC draws its principles from several areas: First and primary is activity theory. Activity theory has been proposed as a unifying theoretical framework for HCI by Nardi in [4]. Activity theory influenced studies of practice and peripheral participation [3, 1] which are important for our work as well. Activity theory divides human behavior into a hierarchy. At the highest level are activities. An activity is an ongoing task, like writing a book or musical score etc, working on a legal case, or building a house. Below activities are actions. Actions include working on the book on a word processor, sitting at a piano to write a score, doing a LEXUS/NEXUS search on behalf of a client, or framing some rooms in the house. What tends to be stable in an activity is the group of people involved, the totality of tools relevant to the activity (although the tool in use varies with the action), and the role of each person in the activity. Importantly, activities are usually social processes while actions are often individual. Computing systems today are organized around actions: the word processor, spreadsheet, database etc. Activity-Based computing gives priority to activities and their constituents.

An Activity-Based Desktop



We start with a conceptual sketch of an activity-based interface, as shown above. The activities are clusters of users, documents and tools (magnifying glass icons). Activity-based computing does not assume this interface, and may not make use of it at all. But the interface will make the following discussion concrete. The reader will notice that it has the following attributes:

Entitys The elements of the display (which are the main entitys in the knowledge ecology) are (i) people (ii) documents, collections or databases and (iii) tools. A complete implementation is likely to make finer distinctions, e.g. people in an activity often fulfil different roles, and they will have different representations on the display.

Grouping into Activities The entities in the display are grouped together into visual clusters that correspond to the user's activities. Each activity will include representations of other participants in the activity, and documents and tools that one or more of the users have used in the activity.

Awareness Levels The display is on a tilted plane. The plane naturally shows objects in the foreground at greater resolution. Object size is graded with depth, allowing more clusters to fit at greater depth. Depth in the display is a concrete representation of level of awareness, i.e. sensitivity to changes in the objects. Awareness decreases with distance.

Centering and Movement One activity is normally in the center of the foreground, and represents the user's current activity. As the user shifts to a different activity, their viewpoint moves over the plane until the second activity is foregrounded. For the visual representation, we anticipate using a hyperbolic or fisheye projection to keep all activities in view.

An ABC system is most useful if it is used by many people. Its robustness, accuracy and value depend on data gathered from many participants in each activity. Since few people can invest time giving indirect help to others, data gathering and analysis for ABC must be highly automated. We conceived the visualization above as a fully automated display. Tacit data mining will be used to infer the activity clusters, the user's current activity from moment to moment, and other important data such as user expertise and document authority. The more automation there is, the less demand on human attention.

Context Sharing

One of the key features of ABC is that it defines an "activity" as the basic shared context between a group of people. An activity is circumscribed by the people or groups involved in it, the documents they use, and the tools needed for the activity. The current activity (the computer's version) is an approximation to the user's current high-level context. Consider the following questions: (i) Can you send this draft to James? (ii) How many motors did Dan order? (iii) What happened to those lenses I ordered from Melles-Griot?

The AB view groups items in an activity together. e.g. To send a memo to James, I could drag a doc icon within the activity cluster to his icon, or select his icon to send him a message. So to figure out how many motors were ordered, I would select Dan and then the "purchasing dept. tool". Representing the purchasing dept using a tool is natural in this context since the AB view of tools is as "mediators". Purchasing is indeed a mediator between the person ordering and a vendor. Similarly, for lenses I begin by selecting myself as the actor, then either Melles-Griot as the object of the action, or purchasing as the tool, or both. ABC includes context relevant to the subject and object of an action, and could potentially be very valuable for contextual disambiguation of speech (e.g. in the 3 examples above).

Document Context

Whereas natural dialogue involves a speaker, listener and shared context, documents have an author(s), a supposed target readership, and a context which applies only to their creation. The disparity between backgrounds of the real and assumed reader, and the differences in creation and reading context can make document interpretation very difficult. Documents have been characterized as "immutable mobiles" [2, 5], that is as objects that move between contexts, but do not adapt to them. The context that should be contained in a "rich" document should include:

Authorship Who were the authors? Identity and job title is not enough. It is important to know what relevant experience the authors have (to the activity) and what their level of expertise is. We are exploring algorithms for estimating expertise from tacit data.

Dependencies on other documents It is important to know what other documents, databases, charts etc. were influential in the current document. Knowing all the documents consulted gives the reader little idea about which were most *influential*. A more useful measure would be tacit, e.g. time the participants spent reading a given document during the activity.

Roles of Authors As well as the expertise of authors, it is important to know the roles of authors, and their level of participation. We rely on ideas from social networks to infer influence as well as power in the group's interactions.

[1] Seth Chaiklin and Jean Lave, editors. *Understanding Practice: Perspectives on Activity and Context*. Cambridge University Press, 1993.

[2] B. Latour and S. Woolgar. *Laboratory life: the construction of scientific facts*. Princeton University Press, Princeton, NJ, 1991. Intro by J Salk.

[3] Jean Lave and Etienne Wenger. *Situated Learning: Legitimate Peripheral Participation*. Cambridge University Press, Cambridge, England, 1991.

[4] Bonnie Nardi. *Context and Consciousness*. MIT Press, 1997.

[5] Susan Leigh Star, editor. *Ecologies of Knowledge: World and Politics in Science and Technology*. State University of NY Press, 1995.