



The Organizational Contexts of Development and Use

JONATHAN GRUDIN

Information and Computer Science Department, University of California, Irvine (grudin@ics.uci.edu)

Most computers are used in organizational settings. However, the 1980s provided an interval in which human-computer interaction research and design could ignore social context and focus on perceptual and cognitive aspects of use. IBM PCs and Macintoshes were difficult to network. They were used in social and organizational contexts, but their use was individual. Results were typically shared as printed hardcopy. In this decade, software engineers, human factors engineers, graphic and industrial designers, and psychologists addressed the perceptual and cognitive challenges of presenting information and conducting human-computer dialogues.

As PCs lost their “insularity,” social and organizational effects grew in design relevance. First came file sharing and file transfer, then email and the Internet. Client-server computing, groupware, and other technologies reflect these changes. As “PC islands” disappear, computing becomes inextricably interwoven with its group and organizational contexts. Over time, the proportion of computers used in engineering settings decreases. Developers of interactive systems must work harder to understand the contexts of use.

As the field of human-computer interaction expands its focus to consider organizational context, we find that this is not wholly unexplored territory. Particularly in Europe, less influenced by personal computer software development, organizational analysis has been a continuous thread in human factors and ergonomics research and practice. In general, the information systems field (also called data processing or manage-

ment information systems) has focused on the organizational effects of computing (see Friedman [1989] and Hirschheim et al. [1995] for historical surveys).

This work focuses on organizations. Between the individual focus of much HCI work and the organizational level lie yet other levels of organization, such as groups and projects. Recent research into group dynamics, much of it by anthropologists or ethnographers who study technology use in modern organizations, is found in the computer-supported cooperative work conferences (North American and European conference proceedings available from ACM and Kluwer, respectively) and the Kluwer journal of the same name.

Organizational context plays two roles in interactive systems development:

- (1) System or application use takes place in organizations. Organizational context shapes the expectations, prior experiences, priorities, preferences, and specific tasks of users. This is of consequence to designers or developers, as well as to those marketing, acquiring, introducing, and using systems. Developers who do not understand the organizational contexts of use increasingly risk failure. Word processors and spreadsheets were initially used in a relatively context-independent fashion, but computer use today is more interdependent.
- (2) Systems development takes place in organizational contexts. These contexts greatly shape the expectations, experiences, and priorities of developers. They affect the resources and

constraints operating on development. Of particular significance for interactive systems development, organizational context can fundamentally affect the possibilities for developer and user interaction.

Organizational context of use. Formal efforts to model aspects of organizations include the software engineering techniques of data modeling or information modeling. Modeling consists of tracking information flow through an organization. These techniques were developed when systems were not interactive and when few resources could be devoted to the human-computer interface. In this setting, identifying “user requirements” meant identifying the information that a user accessed.

By the mid-1970s, structured analysis and structured design methods appeared. These elaborate processes might include “establish human-computer interface” as a sub-phase, or might completely ignore interface development. Integrating usability practices into these formal methods has been undertaken. (Lim and Long [1994] present one such effort and in Chapter 7 review others). These efforts identify organizational, task, and interaction levels, where “task” refers to a high-level work activity and “task analysis” is an *organizational* task analysis. In contrast, much human-computer interaction has focused on *cognitive* task analyses of low-level activities such as cutting and pasting text.

These efforts to wed organizational usability methods to formal software engineering approaches are undertaken to advance HCI beyond being an unstructured collection of techniques. This approach assumes developers are willing to adopt formal methods, and also assumes that pertinent organizational knowledge can be made explicit. Other approaches to bringing organizational knowledge into design include participatory design methods, which bring developers and users together to build implicit awareness of context on both

sides; ethnographic studies, which supplement information modeling approaches by stepping back from specific work processes to examine settings as wholes; and contextual interviewing, an ambitious approach that follows observation and interviewing with analysis (Holtzblatt and Jones, in Schuler and Namioka [1993]).

Organizational context of development. Many aspects of development organizations affect the development process and product: size, geographic dispersal, market, competition, and so forth. Perhaps most significant are the inherent relationships among users and developers. Three major contexts illustrate the effects of organizational purpose and structure [Grudin 1991a]:

- (1) In competitive contract development, users and developers are in distinct organizations. They are typically geographically separated, face legal barriers to communication, and may find it impractical to renegotiate a contract to exploit user feedback. Much design occurs for inclusion in a request for proposals, before the developers are identified. Winkler and Buie [1995] summarize the challenges in this environment and outline strategies to meet them, primarily through raising management awareness.
- (2) In internal or in-house development, barriers to communication can exist, but having users and developers in the same organization and identifiable from the outset creates possibilities for interaction. From this context arose techniques such as sociotechnical design, the Scandinavian participatory or collaborative design, and JAD, all aimed at bringing greater organizational knowledge into development. (See Schuler and Namioka [1993] and the June 1993 issue of *Communications of the ACM*.)
- (3) Commercial product or package development differs from either of the

preceding. Again, users and developers are separated, creating difficulties for user involvement in design. Developers are engaged early but encounter obstacles, often including strong time constraints. Most work in HCI reflects this context. Organizational issues affecting product development are explored in depth in Grudin [1991b].

This list is not exhaustive. A development context of growing significance is custom development, which incorporates elements of each of the three contexts described. A contract may be involved, but an ongoing relationship of development and user organizations is often assumed, and developers may adapt a system for multiple customers.

Only by understanding the context can developers accurately identify possibilities for action, the true source of barriers, and appropriate approaches.

REFERENCES

- FRIEDMAN, A. L. 1989. *Computer Systems Development: History, Organization and Implementation*. John Wiley, New York.
- GRUDIN, J. 1991a. Interactive systems: Bridging the gaps between developers and users. *IEEE Computer* 24, 4, 59–69.
- GRUDIN, J. 1991b. Systematic sources of suboptimal interface design in large product development organizations. *Human-Computer Interaction* 6, 2, 147–196.
- HIRSCHHEIM, R., KLEIN, H. K., AND LYYTINEN, K. 1995. *Information Systems Development and Data Modeling: Conceptual and Philosophical Foundations*. Cambridge University Press, Cambridge, UK.
- LIM, K. Y. AND LONG, J. 1994. *The MUSE method for usability engineering*. Cambridge University Press, Cambridge, UK.
- SCHULER, D. AND NAMIOKA, A. (EDS.) 1993. *Participatory Design: Principles and Practices*. Lawrence Erlbaum, Hillsdale, NJ.
- WINKLER, I. AND BUIE, E. 1995. HCI challenges in government contracting. *SIGCHI Bull.* 27, 4, 35–37.