Steps Toward an Ecology of Infrastructure: Design and Access for Large Information Spaces

Susan Leigh Star • Karen Ruhleder

Graduate School of Library and Information Science, University of Illinois at Urbana-Champaign, 501 East Daniel Street, Champaign, Illinois 61820 and Institute for Research on Learning, Palo Alto, California 94025

S. Star: star@alexia.lis.uiuc.edu
K. Ruhleder: ruhleder@alexia.lis.uiuc.edu

We analyze a large-scale custom software effort, the Worm Community System (WCS), a collaborative system designed for a geographically dispersed community of geneticists. There were complex challenges in creating this infrastructural tool, ranging from simple lack of resources to complex organizational and intellectual communication failures and tradeoffs. Despite high user satisfaction with the system and interface, and extensive user needs assessment, feedback, and analysis, many users experienced difficulties in signing on and use. The study was conducted during a time of unprecedented growth in the Internet and its utilities (1991–1994), and many respondents turned to the World Wide Web for their information exchange. Using Bateson's model of levels of learning, we analyze the levels of infrastructural complexity involved in system access and designer-user communication. We analyze the connection between systems development aimed at supporting specific forms of collaborative knowledge work, local organizational transformation, and large-scale infrastructural change.

(Infrastructure; Collaboratory; Organizational Computing; Participatory Design; Ethnography; Internet; Scientific Computing)

*"An **electronic community system** is a computer system which encodes the knowledge of a community and provides an environment which supports manipulation of that knowledge. Different communities have different knowledge but their environment has great similarities. The community knowledge might be thought of as being stored in an electronic library."* (Schatz 1991, p. 88)

"Does virtual community work or not? Should we all go off to cyberspace or should we resist it as a demonic form of symbolic abstraction? Does it supplant the real or is there, in it, reality itself? Like so many true things, this one doesn't resolve itself to a black or a white. Nor is it gray. It is, along with the rest of life, black/white. Both/neither." (John Perry Barlow 1995, p. 56)

1. What Is Infrastructure?

People who study how technology affects organizational transformation increasingly recognize its dual, paradoxical nature. It is both engine and barrier for change; both customizable and rigid; both inside and outside organizational practices. It is product and process. Some authors have analyzed this seeming paradox as *structuration*: (after Giddens)—technological rigidities give rise to adaptations which in turn require calibration and standardization. Over time, structure-agency relations re-form dialectically (Orlikowski 1991, Davies and Mitchell 1994, Korpela 1994). This paradox is integral to large scale, dispersed technologies (Brown

1047-7047/96/0701/0111$01.25

and Duguid 1994; Star 1991a, 1994). It arises from the tension between local, customized, intimate and flexible use on the one hand, and the need for standards and continuity on the other.

With the rise of decentralized technologies used across wide geographical distance, both the need for common standards and the need for situated, tailorable and flexible technologies grow stronger. A lowest common denominator will not solve the demand for customized possibilities; neither will rigid standards resolve the issue (Trigg and Bødker 1994). It is impossible to have "universal niches"; one person's standard is in fact another's chaos. There are no genuine universals in the design of large-scale information technology (Star 1991a, Bowker 1993).

Furthermore, this simultaneous need for customization and standardization is not geographically based nor based on simple group-membership parameters. An individual is often a member of multiple communities of practice which use technologies differently, and which thus have different demands on their flexible-standard requirements. There is no absolute center from which control and standards flow; as well, no absolute periphery (Hewitt 1986). Yet some sort of infrastructure is needed.

We studied the building of a geographically dispersed, sophisticated digital communication and publishing system for a community of scientists. The system-building effort, which was itself an attempt to enhance and create infrastructural tools for research, took place during a period of immense, even radical change in the larger sphere of electronic information systems (1991–1994). One purpose of the development effort was to transform local laboratory organization, and minimize inefficiencies of scale with respect to knowledge and results. The vision was of a kind of supra-laboratory stretched over the entire scientific community. The needs for both standards and customizable components were equally strong. The system development process also became an effort to bring together communities of practice with very different approaches to computing infrastructure. Designers and users faced two sorts of challenges in developing the system: communicating despite very different practices, technologies and skills; and keeping up with changes occasioned by the growth of the Internet and tools like Gopher and Mosaic. Trying to develop a large-scale information infrastructure in this climate is metaphorically like building the boat you're on while designing the navigation system *and* being in a highly competitive boat race with a constantly shifting finish line.

This paper is about that experience, and about its ultimate failure to produce the expected organizational and infrastructural changes. It offers an analytic framework and vocabulary to begin to answer the question: what is the relationship between large scale infrastructure and organizational change? Who (or what) is changer, and who changed? We begin with a definition of infrastructure, and then focus on two aspects of the system development effort: communication and mutual learning between designers and users.

1.1. When is an Infrastructure?

"What can be studied is always a relationship or an infinite regress of relationships. Never a 'thing.' "—Gregory Bateson

Yrjö Engeström, in his "When Is a Tool?," answers the implied title question in terms of a web of usability and action (1990). A tool is not just a thing with pre-given attributes frozen in time—but a thing becomes a tool in practice, for someone, when connected to some particular activity. The article is illustrated by a photo of a physician working at a terminal covered with yellow post-it notes, surrounded by hand-scribbled jottings, talking on the phone—a veritable heterogeneous "web of computing" (Kling and Scacchi 1982). The tool emerges in situ. By analogy, infrastructure is something that emerges for people in practice, connected to activities and structures.

When, then, is an infrastructure? Common metaphors present infrastructure as a substrate: something upon which something else "runs" or "operates," such as a system of railroad tracks upon which rail cars run. This image presents an infrastructure as something that is built and maintained, and which then sinks into an invisible background. It is something that is just there, ready-to-hand, completely transparent.

But such a metaphor is neither useful nor accurate in understanding the relationship between work/practice and technology. It is the image of "sinking into the background" that concerns us. Furthermore, we know that such a definition will not capture the ambiguities of usage referred to above: e.g., without a Braille terminal, the Internet does not work to support a blind person's communication. And for the plumber, the waterworks system in a household connected to the city water system is target object, not background support. Rather, following Jewett and Kling (1991), we hold that infrastructure is a fundamentally relational concept. It becomes infrastructure in relation to organized practices. Within a given cultural context, the cook considers the water system a piece of working infrastructure integral to making dinner; for the city planner, it becomes a variable in a complex equation. Thus we ask, *when*—not *what*—is an infrastructure.

Analytically, infrastructure appears only as a relational property, not as a thing stripped of use. Bowker (1994) calls this "infrastructural inversion," a methodological term, referring to a powerful figure-ground gestalt shift in studies of the development of large scale technological infrastructure (Hughes 1983, 1989). The shift de-emphasizes things or people as simply causal factors in the development of such systems; rather, changes in infrastructural relations become central. As we learn to rely on electricity for work, our practices and language change, we are "plugged in" and our daily rhythms shift. The nature of scientific and aesthetic problems shift as well. As this infrastructural change becomes a primary analytic phenomenon, many traditional historical explanations are inverted. Yates (1989) shows how even so humble an infrastructural technology as the file folder is a central factor in changes in management and control in American industry. In the historical analysis, the politics, voice and authorship embedded in the systems are revealed—not as engines of change, but as articulated components of the system under examination. Substrate becomes substance.

With this caveat, infrastructure emerges with the following dimensions:

- *Embeddedness*. Infrastructure is "sunk" into, inside of, other structures, social arrangements and technologies;
- *Transparency*. Infrastructure is transparent to use, in the sense that it does not have to be reinvented each time or assembled for each task, but invisibly supports those tasks;
- *Reach or scope*. This may be either spatial or temporal—infrastructure has reach beyond a single event or one-site practice;
- *Learned as part of membership*. The taken-for-grantedness of artifacts and organizational arrangements is a *sine qua non* of membership in a community of practice (Lave and Wenger 1992; Star, in press). Strangers and outsiders encounter infrastructure as a target object to be learned about. New participants acquire a naturalized familiarity with its objects as they become members;
- *Links with conventions of practice*. Infrastructure both shapes and is shaped by the conventions of a community of practice, e.g. the ways that cycles of day-night work are affected by and affect electrical power rates and needs. Generations of typists have learned the QWERTY keyboard; its limitations are inherited by the computer keyboard and thence by the design of today's computer furniture (Becker 1982);
- *Embodiment of standards*. Modified by scope and often by conflicting conventions, infrastructure takes on transparency by plugging into other infrastructures and tools in a standardized fashion.
- *Built on an installed base*. Infrastructure does not grow *de novo*; it wrestles with the "inertia of the installed base" and inherits strengths and limitations from that base. Optical fibers run along old railroad lines; new systems are designed for backward-compatibility; and failing to account for these constraints may be fatal or distorting to new development processes (Monteiro, et al. 1994);
- *Becomes visible upon breakdown*. The normally invisible quality of working infrastructure becomes visible when it breaks; the server is down, the bridge washes out, there is a power blackout. Even when there are back-up mechanisms or procedures, their existence further highlights the now-visible infrastructure.

The configuration of these dimensions forms "an infrastructure," which is without absolute boundary on a priori definition (Star 1989a and b). Most of us, in speaking loosely of infrastructure, mean those tools which are fairly transparent for most people we know about, wide in both temporal and spatial scope, embedded in familiar structures—like power grids, water, the Internet, airlines. That loose talk is perfectly adequate for most everyday usage, but is dangerous when applied to the design of powerful infrastructural tools on a wide scale, such as is now happening with "national information infrastructures." Most importantly, such talk may obscure the ambiguous nature of tools and technologies for different groups, leading to de facto standardization of a single, powerful group's agenda. Thus it contributes to Kraemer and King's "politics of reinforcement" in computerization (1977). Such talk may also obscure the nature of organizational change occasioned by information technology development.

If we add these dimensions of infrastructure to the dual and paradoxical nature of technology, our understanding deepens. In fact, the ambiguity and multiple meanings of usage marks any real functioning system. *An infrastructure occurs when the tension between local and global is resolved.* That is, an infrastructure occurs when local practices are afforded by a larger-scale technology, which can then be used in a natural, ready-to-hand fashion. It becomes transparent as local variations are folded into organizational changes, and becomes an unambiguous home—for somebody. This is not a physical location nor a permanent one, but a working relation—since no home is universal (Star, in press).

The empirical data for this paper come from our work as ethnographers/evaluators of a geographically dispersed virtual laboratory or "collaboratory" system meant to link the work of over 1,400 biologists (Star 1991b). The system itself appeared differently to different groups—for some it was a set of digital publishing and information retrieval tools to "sit upon" already-existing infrastructure; for others it supported problem-solving and information sharing; for yet others, it was a component of an established set of practices and infrastructural laboratory tools. The target users had vastly differing resources and computing skills and relationships, and these in turn were sharply different from those of the designers.

As well, it is increasingly clear to us that this development effort is taking place at a moment of rare, widespread infrastructural change. With the growth of the Internet/World Wide Web and their utility softwares (such as Mosaic, Netscape, Gopher, WAIS), as well as the myriad of email uses, electronic bulletin boards and listservs, the boundaries of system implementation are embedded in the eye of an informational and organizational hurricane of change. For a few of our respondents, the system became a working infrastructure; others, however, turned to Gopher and Mosaic and other Internet tools. And of course, the skill base and learning curve, as well as other factors such as support networks in organizations which help users with such tools, is itself constantly changing. This changing environment, combined with the complexities of implementation from the user's perspective, contributed to the system's ultimate failure in achieving its original goal of becoming the central information resource and the primary communication conduit within a particular scientific community.

2. The Worm Community System (WCS): Background

The Worm Community System (WCS) is a customized piece of software meant to support the collaborative work of biologists sequencing the gene structure, and studying other aspects of the genetics, behavior and biology of *c. elegans*, a tiny nematode (Schatz 1991, Pool 1993). It is one example of a new genre of systems being developed for geographically dispersed collaborative scientific work. WCS is a distributed "hyperlibrary," affording informal and formal communication and data access across many sites. It incorporates graphical representations of the physical structure of the organism; a periodically updated genetic map; formal and informal research annotations (thus also functioning as an electronic publishing medium); directories of scientists; a thesaurus of terms linked with a directory of those interested in the particular subtopic, and a quarterly newsletter, the *Worm Breeder's Gazette*. It also incorporates an independently developed database, *acedb*. Many parts of the system are hypertext-linked.

Its principle designers were computer scientists, some with backgrounds in biology. However, WCS was developed with the close cooperation of several biologists; user feedback and requests from those biologists were initially incorporated into the system over the years of development. Its development was part of a broader project to both construct and evaluate the implementation and impact of a scientific collaboratory. Two ethnographers, Star and Ruhleder, were members of the project team, but not part of the technical development effort per se. The ethnographic component of the project is described in more detail below.

The community consists, as we have stated, of about 1,400 scientists distributed around the world in some 120 laboratories (as of 1994). They are close-knit and consider themselves extremely friendly, as indeed we found them to be. Until recently, most people were first or second "generation" students of the field's founders. Recently, *c.elegans* was chosen as the "model organism" for the Human Genome Initiative (HGI), said to be the largest scientific project in history. "Model organism" means both that the actual findings from doing the worm biology and genetics will be directly of interest to human geneticists, for example when homologues are found between oncogenes (cancer-causing genes) in the worm and in the human (although worms do not get cancer as such, there are developmental analogies). In addition, the tools and techniques developed in the *c.elegans* mapping effort will be useful for the human project.

Senior biologists are concerned that the impact of the HGI and increasing interest in the worm will adversely affect the close, friendly nature of relationships in the community. Viewing this community as a loosely-coupled organization whose members often work in and interact with more formal organizations, these new constraints and opportunities threatened to upset traditional linkages and a collaborative culture heavily dependent on apprenticeship and continued personal contact. Members of the community themselves were willing to become a "model organism" for the ethnographers because they hoped the system would help maintain the community's strong bonds and friendly character in the face of rapidly increasing visibility and growth. In that sense, the goal was not only organizational transformation in terms of available resources and information-sharing opportunities, but also the retention of desired characteristics in the face of transformation for the worse.

The work of *c.elegans* biologists can be captured by the notion of solving a jigsaw puzzle in four dimensions, across considerable distance (the labs we studied were located in the US and Canada; input comes from Europe, Japan and Australia as well). In addition to the four dimensions, the data are structured differently and must be mapped across fields, for example, a behavioral disorder linked with one gene must be triangulated with information from corresponding DNA fragments. Labs working on a particular problem, e.g. sperm production, are in frequent contact with each other by phone, FAX and e-mail, exchanging results and specimens.

The worm itself is remarkable both as an organism, and as a component of a complex pattern of information transfer integral to the biologists' work. It is microscopic and transparent (thus easier to work with than opaque creatures such as humans!). It is a hardy creature, and may be frozen, mailed to other labs via UPS, thawed out and retrieved live for observation. Worms and parts of worms travel from one lab to another as researchers share specimens. Worm strains with particular characteristics, such as a mutation, may be mailed from a central Stock Center to labs requesting specimens. Tracking the location and characteristics of organisms thus is an important part of record-keeping and information retrieval.

Computing use and sophistication in the labs varies widely. In the labs most active in trying out WCS, there are 1 to 2 active, routine WCS users. In many, computing is confined to e mail, word processing, or the preparation of graphics for talks. In most labs there is one "computer person," often a student, who is in charge of ordering new programs and designing databases to keep track of strains and other information.

Our role in the project as ethnographers has been to travel to worm labs, interview about and observe both the use of computing and WCS, and other aspects of routine work, as well as to ask questions on topics including careers in the community, competition, routine information-sharing tasks, how computing infrastructure is managed, etc. We did semi-structured interviews and observations at 25 labs with more than 100 biologists over a three year period (1991–1995),[1] and fed back to designers both specific suggestions ("so-and-so found a bug") and general observations ("such-and-such would violate community norms"), several of which were incorporated into development.

Sociological analysis to support computer design is relatively new (Bucciarelli 1994). The participatory design approach developed in Scandinavia paved the way for workplace studies which inform design (Ehn 1988, Bødker 1991, Anderson and Crocca 1993), usually using

[1] Names have been changed to preserve anonymity.

a combination of a case study approach and action research, with rapid feedback from users of computing systems. Where possible, we adapted those principles. At the same time, trying to cover a geographically distributed community in aid of complex systems development also meant that neither a strict case study nor rapid prototyping were possible. We covered as much territory as possible with traditional interviewing and observational techniques. The analysis of the data was conducted with a grounded theory approach, beginning with a substantive description of the community and moving to more abstract analytical frameworks as our comparative sites grew in number (Strauss 1986).

Most respondents said they liked the system, praising its ease of use and its understanding of the problem domain. On the other hand, most have not signed on; many have chosen instead to use Gopher and Mosaic/Netscape and other simpler net utilities with less technical functionality. Obviously, this is a problem of some concern to us as system developers and evaluators. *Despite good user prototype feedback and participation in the system development, there were unforeseen, complex challenges to usage involving infrastructural and organizational relationships.* The system was neither widely adopted, nor did it have an immediate impact on the field as the resources and communication channels it proffered became available through other (often more accessible) means. However, the WCS itself continues to change and adapt; the latest version is based entirely on Web technology, and the Web will shortly have enough functionality to reproduce the custom software WCS.[2]

3. Signing On and Hooking Up

Those working in the emergent field of Computer-Supported Cooperative Work (CSCW), of which the collaboratory is a subset, have struggled to understand how infrastructural properties affect work, communication, and decision making (Kraemer and King 1986; Schmidt and Bannon 1992; Malone and Olson, in press).

One of the classic CSCW typologies has distinguished important task differences for synchronous/asynchronous systems; proximate/long distance use; and dedicated user groups vs. distributed groups with fluctuating membership (Ellis et al. 1991). This was useful for characterizing an emerging group of technologies; however, it offers no assistance in analyzing the issues associated with implementation or integration (Schmidt and Bannon 1992). It also does not analyze the relational aspects of computing infrastructure and work, either real time "articulation work" or aspects of longer-term, asynchronous production tasks. We encountered many such issues in the worm community in the process of "signing on" and "hooking up" to WCS—tasks related to finding out about the system, installing it, and learning to use it. For most of the worm biologists we interviewed, the tasks involved in signing on and hooking up had preoccupied them, and they had not gotten over the initial hurdle and into routine use.

Consider the set of tasks associated with getting the system up and running. WCS runs on a Sun Workstation as a standalone or remotely, or on a Mac with an ethernet connection remotely over the NSFnet, or, with less functionality, on a PC over the net. Prior to using WCS, one must buy the appropriate computer; identify and buy the appropriate windows-based interface; use a communications protocol such as telnet and/or FTP; and locate the remote address where you "get" or operate the system. Each of these tasks requires that people trained in biology acquire skills taken for granted by systems developers. The latter have interpersonal and organizational networks that help them obtain necessary technical information, and also possess a wealth of tacit knowledge about systems, software, and configurations. For instance, identifying which version of X Windows to use on a workstation means understanding what class of software product X Windows is, installing it, and then linking its configuration properly with the immediate or remote link. Following instructions to "download the system via FTP" requires an understanding of file transfer protocols across the Internet, knowing which issue of the *Worm Breeder's Gazette* lists the appropriate electronic address, and knowing how FTP and X Windows work together.

These common issues of shopping, configuration, and installation are faced in some degree by all users of computing. But solving these "shopping" and informational issues will not always suffice to get work done smoothly. For instance, deciding to buy a SPARC station (one popular UNIX-based workstation) and run it

[2] Personal communication between Star and Bruce Schatz, 9/28/95.

on a campus which has standardized itself on DOS machines may bring you into conflict with the local computer center, and their attempts to limit the sorts of machines they will service. Or there may be enough money to buy the computer, but not enough to support training for all lab staff; in the long term, this disparity may create inequities.

We discovered many such instances, common to a variety of system development efforts and types of users, and all interesting for the design of collaborative systems. With the advent of very large scale systems such as the US National Information Infrastructure, they become pressing questions of equity and justice, as well as questions of organizational formation, *trans*formation, and demise. They simultaneously enact technological infrastructure and social order. We encountered a myriad of contexts and tasks surrounding system use. These varied in complexity and consequences, and we borrowed a metaphor from learning theory to characterize these variations.

4. Levels of Communication and Discontinuities in Hierarchies of Information

The "tangles" encountered in signing on and hooking up occur in many venues, and may inhibit desired organizational transformation; at the least, they inform its character and flavor the growth of infrastructure. We turned to Gregory Bateson as a theorist of communication for a more formal understanding of the ways in which communicative processes are entangled in the development of infrastructure. We rely on his *Steps Toward an Ecology of Mind* (1978). The term ecology, as adapted to our analysis here, refers to the delicate balance of language and practice across communities and parts of organizations; it draws attention to that balance (or lack of it). It is not meant to imply either a biological approach or a closed, functional systemic one.

4.1. Bateson's Model

Bateson (1978), following Russell and Whitehead, distinguishes three levels in any communicative system. At the first level are straightforward "fact" statements, i.e. "the cat is on the mat." A discontinuous shift in context occurs as the statement's object is changed to "I was lying when I said 'the cat is on the mat'." This second order statement tells you nothing about the location of the cat, but only something about the reliability of the first order statement. In Bateson's words:

> "There is a gulf between context and message (or between metamessage and message) which is of the same nature as the gulf between a thing and the word or sign which stands for it, or between the members of a class and the name of the class The context (or metamessage) *classifies* the message, but can never meet it on equal terms " (p 249)

At the third level, the gulf appears in evaluating the context itself: "There are many conflicting approaches to evaluating whether or not you were lying about the cat and the mat." In this sentence, the listener's attention is forced to a wider and deeper range of possibilities; again, it may classify the message about lying, but is of a different character.

Theorizing the gulf between levels, Bateson and others have gone on to classify levels of learning with similar distinctions and discontinuities. There is a first and second order difference in learning something and learning about learning something; and between the second and third are even more abstract differences between learning to learn, and learning about theories of learning and paradigms of education. As the epigraph to an earlier section indicates, of course the regress upwards is potentially infinite.

For our purposes we identify three levels (or "orders") of issues that appear in the process of infrastructure development, and discuss each with respect to the worm community and WCS. As with Bateson's levels of communication or learning, the issues become less straightforward as contexts change. This is not an idealization process (i.e., they are not less material and more "mental"), nor even essentially one of scope (some widespread issues may be first order), but rather questions of *context*. Level one statements appear in our study: "Unix may be used to run WCS." These statements are of a different character than a level two statement such as "A system developer may say Unix can be used here, but they don't understand our support situation " At the third level, the context widens to include theories of technical culture: "Unix users are evil—we are Mac people." As these levels appear in developer-user communication, the nature of the gulfs between levels is important.

First-order issues may be solved with a redistribution or increase of extant resources, including information. Examples would be answers to questions such as: What is the e mail address of WCS? How do I hook up my SPARC station to the campus network?

Second-order issues stem from unforeseen or unknowable contextual effects, perhaps from the interaction of two or more first-order issues. An example here is given above: what are the consequences of my choosing a SPARC station instead of a Mac, if my whole department uses Macs? If I invest my resources in learning WCS, are there other more useful programs I am neglecting?

Third-order issues are inherently political or involve permanent disputes. They include questions about schools of thought of biological theory for designing the genetic map of the organism for WCS. They raise questions such as whether competition or cooperation will prove more important in developing systems privacy requirements, and whether complexity or ease of use should be the main value in interface design. Such questions may arise from an interaction of lower order issues, such as the choice of computer system and the tradeoffs between scientific sophistication and ease of learning.

In this sense, infrastructure is context for both communication and learning within the web of computing (Kling and Scacchi 1982). Computers, people, and tasks together make or break a functioning infrastructure. In Bateson's words:

> "It becomes clear that the separation between contexts and orders of learning is only an artifact . The separation is only maintained by saying that the contexts have location outside the physical individual, while the orders of learning are located inside. But in the communicational world, this dichotomy is irrelevant and meaningless . . . the characteristics of the system are in no way dependent upon any boundary lines which we may superimpose upon the communicational map." (p 251)

Information infrastructure is not a substrate which carries information on it, or in it, in a kind of mind-body dichotomy. The discontinuities are not between system and person, or technology and organization, but rather between contexts. Here we echo recent work in the sociology of technology and science which refuses a "great divide" between nature and artifice, human and nonhuman, technology and society (e.g., Latour 1993).

These discontinuities have the same conceptual importance for the relationship between information infrastructure and organizational transformation that Bateson's work on the double bind had for the psychology of schizophrenia. If we, in large-scale information systems implementation, design messaging systems blind to the discontinuous nature of the different levels of context, we end up with organizations which are split and confused, systems which are unused or circumvented, and a set of circumstances of our own creation which more deeply impress disparities on the organizational landscape.

We apply this typology below within the context of "signing on" and "hooking up." Following that application, we discuss the implications of this typology for other forms of information systems development, and the broader implications for understanding the impact of new computer-based media and their integration into established communities.

4.2. First-order Issues

First-order issues are often those which are most obvious to informants, as they tend towards the concrete, and can be addressed by equally concrete solutions (more money, time, training, or support). The first-order issues in this setting center around the installation and use of the system, and include finding out about it, figuring out how to install it, and making different pieces of software work together. First-order issues, however, are not limited to "start-up," but recur over time as work patterns and resource constraints shift (and thus perhaps a by-product of second- or third-order changes).

4.2.1. Informational Issues. Potential users needed to find out about the system and determine the requirements for its installation and use. "Shopping" for the system involved decisions about hardware and software, and sometimes also involved agreements with other departments to share resources or funding; at one major lab, the "worm" people had WCS loaded onto a server owned by the "plant" people on the floor above them; establishing this agreement involved finding out about WCS resource needs and the local availability of these resources. This agreement negated the need to find out about system building and maintenance, since the worm people were piggy-backing off the original efforts of the plant people to purchase and put the server in place.

4.2.2. Issues of Access. In some labs, physical access was critical. WCS might be located in an overcrowded and noisy room, stuck in the corner of a lounge, on a different floor of the building altogether, or accessible only during certain hours. This was the case in the deal cut between the worm and plant people, above: "The WCS and *acedb* are really on a machine upstairs, it belongs to the plant genome project people. . . . We can only use it evenings, weekends" (Brad Thomas, PD).[3]

Other labs experienced time limitations and physical inconveniences: "You can access *acedb* through the Suns downstairs, but it's not convenient. You can only do it after hours. People just won't use it" (Eliot Red, PD) or: "Our computing is good compared to other labs. I finished up a Ph.D. at UCLA, they had one VAX, some PCs, you had to walk to another building to use the VAX" (Brad Thomas, PD).

When we asked whether lab notebooks would one day be replaced with small palmtop computers or digitized pads, researchers were dubious. Respondents at one cramped lab in an urban high-rise simply noted that there was no place to put another computer, even a small one. They shared their lab with another group, and even lacked space for some necessary lab equipment. Such simple spatial or architectural barriers are crucial for the usability of any system, especially those conceived and designed as integral parts of someone's workflow.

4.2.3. Baseline Knowledge and Computing Expertise. Computing expertise was unevenly distributed within the labs; much equipment seemed out of date or unsophisticated. One senior researcher was not aware that databases were available without fixed-length fields, and a PI made a category error in discussing operating systems and applications (equating "a Mac" and "a UNIX"). In general, PIs thought that the level of knowledge was rising through undergraduate and graduate training, but empirically this did not seem to be the case. This might have constituted a learning level gulf (equating the ability to use on-line applications with the ability to understand broader systems concepts). Although there were a few highly skilled people, and one or two with advanced computing expertise, these were not clustered in either the graduate student or postdoc categories.

This sort of knowledge may be an access issue just as much as are space or location. First-order issues in this arena certainly include not only learning to use WCS software, but understanding the platform on which it runs. WCS itself is designed to be extremely user-friendly, and can be effectively used without much difficulty. The typical user in our study was a graduate student, post-doc, or principle investigator with enough knowledge about both domain and community to read a genetic map and recognize the importance of the *Worm Breeder's Gazette*. One user commented: "I just turned it on, pushed buttons" (Ben Tullis, PS).

In fact, at demos and trials at conferences, most users found WCS to be fairly easy and intuitive, once they were on it. However, the platform on which it is based was not transparent (to biologists). WCS runs under UNIX, and both the operating system and software such as X Windows or Suntools requires expertise most biologists didn't have:

> "UNIX will never cut it as a general operating system. Biologists won't use it, it's for engineers [Someone in the lab] had a printing question, took him three months to get something to print" (Bob Gates, GS)

Furthermore, many respondents were unclear about carrying out other sorts of networked computer tasks, such as uploading and downloading files from mainframe to terminal. This made it difficult for them to integrate WCS use with e mail correspondence, word processing files, and other Internet information spaces.

Training often took place in a haphazard way, depending on everything from luck to personal ties:

> "I learned by using it as an editor The second time I learned the formatter. A lot of people are comfortable with e mail, and a lot of people are now using GenBank and sequencing packages We get some on the job training. [Two of the grad students in the lab] write up instruction sheets. The person who was the systems administrator until February was a good friend Got a lot of push and shove from him, a lot of shared ideas" (Jeff Pascal, PD)

[3] PI = Principle Investigator, PD = Post Doc, GS = Graduate Student

No lab offered special training in computing, although some students had taken classes at local computer centers. Several said that they only learn "exactly enough to suit what you have to do" (Carolyn Little, PI).

4.2.4. Addressing First-order Issues. On the surface, these issues may be solved in a fairly straightforward fashion. Effective shopping requires appropriate information, gathered and evaluated by a technically knowledgeable individual. When expertise needed for making computing decisions doesn't reside in the lab itself, it can be brought in from the outside, perhaps by turning to a campus computing facility or hiring a savvy undergraduate in computer science. Proposals can be written for equipment purchases. Issues of physical access can be solved by making the case for additional space. Issues of technical access can be solved by additional training. For instance, just as departments in the humanities are starting to offer tutorials or even certificates in humanities computing, biology departments could offer similar tutorials tailored to the needs of their own communities.

However, first-order selection issues are often intermingled with or converge to form higher-order issues. Shopping, for instance, is not *just* a matter of getting the right information to the right person, but requires information distribution channels that bridge several academic communities: worm biologists, tool builders and local computing support centers. Similarly, when shopping and selection raise questions of standards, they become intermingled with questions of organizational and workplace culture ("Unix is for engineers, not biologists"). This is a particularly salient issue in instances where multiple groups share computing, or where computing support is only available for a limited set of technological choices.

4.3. Second-order Issues

Second-order issues can be analytically seen either as the result of unforeseen contextual effects, such as aversion to UNIX by biologists, or as the collision of two or more first order issues, such as uncertainty during shopping combined with lack of information about how to hook up the system. These sorts of combinations can mean the person is forced to widen the context of evaluation, and link choices about software packages with best guesses about the direction of the organization. Included in this category are cultural influences on technical choices; paradoxes of infrastructure; "near-compatibility" and the "almost-user community"; constraints becoming resources; and understanding the nature of baseline skills and their development. They are second-order because they broaden the context of choice and evaluation of the straightforward first-order issues such as obtaining software and access to machines.

4.3.1. Technical Choices and a Clash of Cultures. Shopping and selection interact not only with training and ease-of-use issues, but with organizational cultural issues. For example, five people independently mentioned being put off by UNIX, usually in the context of comparing it favorably with the Mac. One PI mentioned having no base of UNIX knowledge available from the local computer center, although he had taught himself enough to run a SPARC station (Joe White, PI). Others expressed similar sentiments: "As long as it's easy we'll use anything, like the Macs. So you can do like cut and paste, like you can on the Mac" (Eliot Red, PD) and, "We were previously using UNIX but this is much easier. UNIX is impossible. It's a real pain. This is much easier. The Macs, you know. . . ." (Linda Smith, PI).

One person who defined himself as a "crossover" person (between biology and computing) said,

> "It's a big problem. Biologists are Mac people, and UNIX is an evil word Most people are afraid of it, and refuse to use it. 'If it's not on Mac I don't want it.' There are a lot of problems getting people to use it, rather than delegate the use of it." (Harry Jackson, GS)

Yet UNIX, apart from forming a basis for the WCS and the language of its design team, was often also the language of the computer scientists who supported and maintained the local university computing environment. This apparent gulf between user communities led some biologists to speculate that there are "two types of scientists—love or hate the computer," and that "the only way they'll ever [use] it is by force" (Jeff Pascal, PD). They attributed successful computer use to "some kind of natural affinity" (Eliot Red, PD). This divergence has important implications for training, as do some other basic "cultural" issues.

4.3.2. Paradoxes of Infrastructure. The uneven spread of computing expertise and resources shows vividly how a simple increase, or lack, of first-order resources cannot fully explain a successful infrastructure. Differences of expertise and local organizational savvy between relatively rich and relatively poor labs may override first-order concerns. One of the poorest labs, for example, still running outdated IBM PC-XT equipment, actively used the system, had developed its own databases, and tracked strain exchange with a level of sophistication unparalleled in the community. The lab's PI loved to "play around" with software and hardware, and loved the challenge of overcoming the limitations of his lab, second-order problems were thus reduced to first order by his own skill and interest.

The richest lab, on the other hand, had just received a substantial grant from the Human Genome Initiative to completely "hook up" the entire biology infrastructure on campus. However, they were unable to operate the system through a combination of "waiting for the ethernet" and "waiting for the Sun." The PI illustrates the dilemma:

> "We applied for an ethernet in May (laughs) It should be here [in a few years] . They'll be independent of the building network, [the people] on the SPARC The Macs will be on the building network " (Linda Smith, PI)

A graduate student continues the story:

> "No one will put the wires in, though . we made a deal with the network people [network services] that we'd run wires and they'd connect it up They manage all the campus networks [Someone else] has dealt with Sun, though " (Steve Grenier, GS)

At the time of the interview, they had strung their own cables and were waiting on the delivery of the SPARC stations. Linda Smith (the PI) then anticipated having to spend a lot of time to "get the software underway."

Even institutions with outstanding technical support had no organizational mechanisms for translating that expertise to highly domain-specific questions, applications, and issues. Campus computer centers were often neither knowledgeable about nor interested in applications packages relevant to biologists and geneticists, nor was there support for independently purchased hardware or software:

> "Computing support s**** at [Research Institution]. I called the center for help with installing WCS on the Sun and they basically told me, find a UNIX guy, buy him some pizza If we have problems with the network or programs they support, they do it If you didn't buy your hardware from them, forget it If they don't support your software, forget it. It's handled on a department by department basis Biology has no infrastructure " (Bob Gates, GS)

Who "owns" a problem or application was locally-determined, and attribution of ownership made a great difference in individuals' ability to get help. Some PI's developed on-campus linkages that would bring computing expertise to bear on their own problems. The PI of one small lab submitted a grant together with a computer science faculty member interested in the visualization of scientific data. Together they planned to develop a tool for visual data representation and analysis; in the process, the PI will get not only a tool to support his research, but a UNIX-based workstation from which he can access WCS.

These issues were of great concern to post-docs looking to start up their own labs with increasingly limited funds. WCS was seen as a tool of the "upper tier" of richer labs (Harry Markson, GS), and described as "a rocket" when "we need a Model-T" (Marc Moreau, GS); a post-doc planning to start his own lab complained that:

> "Half a system for everyone is better than a really great system for just a few labs we had to hire [a computer specialist affiliated with another lab] Even the computer guys here [two graduate students] worked on it three weeks, and they couldn't load the [WCS] system It's oriented to big labs " (Jav Emery, PD)

He added, "If it's not on a Mac or IBM, it won't get to people," and suggested, "you need a modular system, you need to be able to have parts of the database running on the Mac, *reach the small labs*" (emphasis added).

4.3.3. Tensions Between a Discipline in Flux and Constraints as Resources. What might be seen as constraints that could be overcome with technology may become resources from a different perspective. We proposed that it would be trivially easy to make *The Worm Breeder's Gazette* available on a continual-update basis. On the one hand, continual updates would have served the needs of a very fast-moving community: "The faster the [WCS] update, the better. . . . You do it through the *Gazette,* you contribute regularly. You're competing [with other labs] on the same gene" (T. Jones, GS).

> "You need frequent updatıng, shortly after each *Gazette,* ı e., within two weeks after a *Gazette* there should be a new release. . . The WCS *Gazette* could replace hard copıes, it would be cheaper." (Brad Thomas, PD)

Yet other respondents objected strongly to this option, even though they worked in the same competitive environment. Objections centered around the utility of community-imposed deadlines on structuring work, both in terms of submitting and reading articles: "I would run the newsletter exactly how it's run now. Just leaving it open ended is not good. If there is infinity there is never a time to review things. And no deadlines" (John Wong GS);

> "If the WCS were used to publish *Gazette* artıcles, what would be optimal? Well, contınual would not be so good. There is something to be said for deadlines. Even sıx tımes a year, and it becomes background noise. . . ıt's hard to predict whether a frequency change will change the ımpact." (Gordon Jackson PI)

The deadline was simultaneously constraint and resource.

The distribution pattern for the *Gazette* affected not only the work habits of individuals, but was integrally linked to communication and coordination within labs and across them:

> "(*Do you thınk the WCS will replace the Gazette?*) If ıt replaces ıt, then we won't read it. I mean, when the *Gazette* arrıves we split it up and each read a part Then we use ıt to get ınto other people's work." (Ed Jones, GS)
>
> "(*What kınds of informatıon do you not keep on the computer?*) Well, you couldn't have the newsletter on electronically. The constant update would be a nıghtmare There would be no referenceable archive." (Paul Green, PI)

This last point is an important one, since the *Gazette* serves as a reference database containing not only pointers to work being carried out in various labs, but to protocols, etc. For newcomers to the discipline, it serves as an important teaching tool; the on-line version would make back issues available more easily. A continual-update format would require a new way of referencing or indexing contributions. One person envisioned a different form of ongoing information service, "something in between a formal and informal database" where "if you have little writeups you could put them in an annotation box" (Alan Merton, PI). As for the *Gazette,* he suggests, ". . . you could put a more formal thing into an on-line *Gazette* format, and keep it as it is" in terms of content, timing, and organization.

4.3.4. "Near-compatibility" and the "Any Day Now" User. Sometimes the gulf between first and second order appears as a sense that what is happening *should* be first order. So strong is this sense that it can lead to some seemingly odd behavior. We encountered a persistent idea among respondents that they were "just about to" be hooked up with the system, and that the barriers to hooking up were in effect trivial. Sometimes this even caused them to say that they *were* using the system, whereas observations and interviews in fact showed that they were not. For instance, when trying to find a site to observe in a large city with several universities and several labs listed as user sites, one of the authors spent almost a week tracking down people who were actually using the system. No one she talked with was using it, but each person knew of someone else in another lab who supposedly was. After following all leads, she concluded that no one was really using the system, though they all "meant to," and figured that it would be available "any day now."

This is not difficult to observe ethnographically, but presents a real difficulty in administering surveys about use and needs. It is clear that this representation is not mendacious, but a common discounting of what seem, from a distance, to be trivial "plug-in" difficulties. The above observations of the difficulties associated with hooking up and getting started, coupled with infrastructural limitations would suggest that these issues are not trivial at all. In fact, these issues turn out to be lethal as they become both chronic and ubiquitous in the system as a whole.

4.3.5. Addressing Second-order Issues. In principal, second-order issues can be resolved by combining an increase in resources with heightened coordination or cooperation between different technical and user communities, such as installing a user support telephone line, hiring a "circuit rider" who can help with hooking up and integration difficulties, and increasing other skill resources locally. However, realistically, biologists are not among the richest of scientists, despite the influx of Human Genome money. Money for capital expenditures is especially scarce, and decisions made about the purchase of or commitment to a particular system often persist for a decade or more. Thus, second-order issues in system use and development may become third-order issues: "why should this lab get resources, which problem is the most important one?" These issues occur at the level of the broader community and transcend the boundaries of any particular institution.

4.4. Third-order Issues

Third-order issues are those which have been more commonly identified by sociology of science in discussions of problem-solving. They have the widest context, involving schools of thought and debates about how to choose among second-order alternatives. These permeate any scientific community, for the reason that all scientific communities are interdisciplinary and contain different approaches and different local histories. They plague communities which are growing rapidly, working in uncharted areas, and which are exceptionally heterogeneous. Third-order issues may not be immediately recognized by members of the community as such, as they can be part of the taken-for-granted. Nevertheless, they have long-term implications. With respect to difficulties of signing on and hooking up, they include triangulation and definition of objects, multiple meanings of information, and network externalities.

4.4.1. Triangulation and Definition of Objects. Different lines of work in the worm community come together in sharing information, including genetics, molecular biology, statistics, etc. One person explained, "I came from [another lab] where I was working on frogs" (Brad Thomas, PD). Another person described himself as "really a developmental geneticist," and adds that a few years ago, "the field was smaller; . . . now many people are coming from outside, from mammals, protein labs" (Harry Markson, GS). Many people moved into the worm community from other areas after graduate school. Differences sometimes fell along the classical lines of organismal biology vs. molecular or genetic research: "I am more of a wormy person. That's true of the community in general. Sometimes you choose a system that's more organismal" (Jane Sanchez, PD).

Collaboration may take place across disciplinary or geographic boundaries:

> [*Are you collaborating with anyone?*] "I'm collaborating with people in the worm and non-worm community Mostly immunologists in the non-worm community, people interested in the immune system In the worm community, I'm collaborating with [a person in another state], on [a particular gene] " (Harry Markson, GS)

Disciplinary origin and current area of work affected the kinds of information individuals needed, and the tools and data sources with which they are familiar. Those studying the organism for its own sake differed in their information needs from those using it exclusively as a model organism; many informants had very specific expectations for WCS data:

> "You need more options, especially for sequencing. This will be especially important once the [Human Genome Initiative] gets underway We need to work with subsets of sequences, examine them in more detail." (Brad Thomas, PD)
>
> "What you'd want is a parts list, a list of cells. . If it's a neuron, its connections with other neurons. . That's for neurobiologists " (Harry Markson, GS)

Identifying the system with a particular subline of work and not as a general utility increased the barriers to usage. System construction was further complicated by another layer of object definition, in that some respondents felt that WCS represented "CS people [computer scientists] . . . building a system only for CS people," and that (WCS) "has a vision that isn't necessarily what biologists want."

4.4.2. Multiple Meanings, Data Interpretation, and Claim Staking. The nature and character of the community was changing as more people entered the "worm world" from other disciplines. During the last decade, it grew from a few hundred to over a thousand members: "It's neat that it's exciting now, but it's also strange to have so many people . . ." (Jane Sanchez, PD/RS)

One goal of WCS is to support communication in a scientific community known for its willingness to share information, but the growth of which has exceeded the ability of informal communication networks to serve as a conduit for this information (Schatz 1991). The issues of developing a collaborative system, however, go far beyond the technical. The multiple meanings or interpretations of particular communications turn out to be important at all levels.

For example, suggestions that it might be useful to have a "who's working on what" directory in the system raised issues of competition and the role of secrecy. Some said they would hesitate to put in certain kinds of information, or wanted announcements delayed until "they had findings":

> "There's always a problem you're going to get scooped You always walk a very fine line There's a lot of people working on my problem if you publish in the *Gazette* you can lay claim to it. People would respect it There have been some clashes, some labs trying to glom on to how much they can It's going to be a struggle from here on out . It's complex with the claim staking. That's why you want to get into it far enough so you can get ahead—before you announce it. If you could preface it with "wild speculation" (laughs) . . well, there's a lot of times those can have a big payoff But then again if five people jump on it, and in the meantime you're scooped that's not so good!" (Mike Jones)

Different communication channels also implied different degrees of freedom: "You can be wrong with no stigma" in the newsletter, said one graduate student, and a PI explained: "People are reluctant to do annotations [to the newsletter]. . . . It's the fear of putting yourself on the line. Making a commitment to what you're doing. It means being wrong in the eyes of your colleagues." (Joe White, PI) He suggested delayed publication of annotations, letting them sit locally for a month or so first, and a post-doc at another lab suggested the implementation of a personal level and a public level of annotation (Brad Thomas, PD). Another PI, however, became angry at this idea. He felt that this would work directly against WCS's commitment to community-wide sharing of information and turn the WCS into a local tool rather than a community resource.

Trust and reliability of information is a final concern worth noting: articles in the *Gazette*, annotations, etc., all carry some kind of implicit value with them. First of all, information ages; old data is superseded by new, problems are resolved, protocols updated. Neither traditional sources nor the current WCS have any fixed way of marking the relative validity or trustworthiness of a set of data. Annotations with updates or the ability to "grey out" old data in the *Gazette* might present technical solutions to these problems, yet there are sometimes no clear-cut answers to these questions, especially in a community populated with multiple viewpoints. In general, says one post-doc, "there is no right or wrong, . . . you have to reach consensus on things, you have to look at labs, which labs you trust more" (Brad Thomas, PD). He wanted to use annotations as a means of raising alternate viewpoints: "I'm knowledgeable [in area X]. Sometimes others who clone don't know as much, they write things that are wrong. I would feel entitled to make annotations." Under the scheme he proposes, it would still be up to the reader of the annotation to sort out and make sense of competing information. He noted wryly that people will cite you as a foil when you've said something incorrect in any event, however, and that there's no way to prevent this. All these instances of data meaning different things under different circumstances—who notifies whom and when, what medium is used, who makes an annotation, or why a particular citation is and isn't included—required knowledge of the community that wasn't captured in any formal system (Star 1989a).

4.4.3. Network Externalities and Electronic Participation: Subtleties and Cautions. The notion of externalities originates in economics and urban planning; a city may be said to afford "positive externalities" of cultural resources. For an artist, New York's externalities usually outweigh those available in Champaign, Illinois, although other amenities such as cost of housing and safety may be greater in the latter. A network externality means that the more actors actively participate in a system or network, the greater the potential, emergent resources for any given individual; it is distinct from the notion of "critical mass," which focuses on the number of subscribers/users at which system use becomes viable. Externalities may be negative in that, eventually, not being "hooked up" may make it impossible to participate effectively within a given community of work or discourse. For instance, the telephone network became a negative externality for those businesses without telephones sometime in the early 20th century; electronic mail has recently acquired a similar status in the academic world. For some purposes, standards (as in information standards) form important aspects of network externalities—i.e., users of nonstandard computing systems are at a disadvantage as network externalities become intertwined with particular operating systems and data interchange protocols (see David (1985) for a cogent analysis and example).

It is currently still difficult to understand the role of network externalities in the worm community, but as electronic access becomes the primary access mechanism for some forms of data, and as participation in all forms of electronic communication rises, they become increasingly important. Let us consider two examples: the WCS as an element of democratization, and, more generally, data repositories as both a means of maintaining openness in the community and a means of providing value to the community.

One goal of the system development is democratization of information—the facilitation of access to critical data through a uniform interface. Yet the more central WCS becomes to the community either as a whole, or as defined by key labs, the more those who cannot sign on along with the others will suffer. The "politics of reinforcement" suggest that the rich labs—either in terms of extant computing infrastructure or in their ability to procure or develop it using internal resources—will get richer as network externalities become more dense (Kraemer and King 1977). This issue may be receding in importance as alternatives to WCS emerge via data available at FTP sites and through Gopher and Mosaic; much of the information available via WCS can now be "pulled from the net." Nevertheless, WCS is superior in its possibilities for graphical representation, and some forms of data analysis require such tools.

Issues of participation persist in several venues. For instance, a key repository is the genetic map, which represents the relative positions of genes on the chromosomes; another is the physical map, which represents cloned fragments of worm DNA and how they overlap to form the chromosomes (Schatz 1991).

> "There's a time problem. You want experts doing this, but you want to do your own stuff, you don't want to maintain a database If you want this to serve a global community, you have to get the data properly defined." (Brad Thomas, PD)

> "There are data that should be on the [physical] map, but they are buried in labs all over the world . When it was fragmented, people sent in clones. Now it's filled in, more coherent. The need to communicate back broke down. There used to be a dialogue, now there's a monologue. They don't bother telling Cambridge they've cloned genes . With the genetic map there's still dialogue." (Ben Tullis, PD)

Some of this is an issue of time; two attempts at an electronic bulletin board for the worm community "died out within two weeks due to lack of contributions" (Bob Gates, GS). Annotation and updating take work, and "it's not of immediate profit" (Sara Wu, PD). However, other reasons were also cited. When asked why the dialogue broke down, the person quoted above replied:

> "There's a communication pyramid You've got approximately 600–700 people in the community [in 1991]. One third of the community arrived between when [the community] was fragmented and [when it was] cohesive They know only the cohesive map " (Ben Tullis, PD)

Newcomers weren't there when these repositories were created, and did not share commitment to their upkeep and growth. Competition was also cited as a factor, and was linked to the issue of timing discussed above. Someone who overheard the question on dialogue breakdown contributed the following comment:

> "Yeah, like [one very well-known] lab, not sending in a note [on X]. And [another well-known] lab, they don't publish things when [they] are close to a gene they're working on." (Kyle Jordan, PD)

A graduate student in the same lab echoes a similar view of data-sharing:

> "Instant updating won't go far People who want an immediate result to be known only want a small number to know. It's more competitive, people are more careful. They don't want everything to be global. By the time it gets into the *Worm Breeder's Gazette*, it's not critical any more The people who really need to know already know " (Bob Gates, GS)

WCS does not maintain the databases or publications featured in these discussions, but it would provide uniform access and an easy-to-use interface to them (once the system is up and running). It derives a significant part of its own value from community participation in their upkeep and maintenance. Without community commitment to the maintenance and upkeep of these materials, WCS has neither value nor legitimacy as a system that fosters either communication or collaboration.

Furthermore, if WCS is to develop its own niche within the community, it will also have to develop its own role in terms of legitimating, documenting, and disseminating information. Currently, for instance, an annotation published in WCS has uncertain value within the community:

> ". . contributing to the WCS, it's not a real publication. You have to send stuff to the *Worm Breeder's Gazette* if you want to publish widely." (Jane Jones, PD)
>
> "You get a better sense of contributing when you send to the *Gazette* If you annotate the WCS, you don't know if it's being read." (Morris Owe, GS)

This is an issue that will face similar systems as they try to piggy-back on established systems, repositories, etc., especially when in competition with multiple other avenues for information retrieval and electronic communication. It is also important in the building of digital libraries and publishing systems—will an electronic journal publication "count" as much as a printed one?

All of us, in addition, face paradoxes of efficiency, or information overload and the danger of diverting a successful manual information tracking system over to the computer with a loss of productivity. Many economists have noted the so-called "productivity paradox" in firms with the introduction of information systems (productivity often declines with investment in IT). Similar paradoxes are a real danger in science with its delicate funding processes, understudied task structures, and fuzzy means of measuring productivity.

4.4.4. Tool Building and the Reward Structure in Scientific Careers. Finally, the role of tool building and tool maintenance may be undergoing a shift as computer-based tools become more prevalent. The tension between traditional notions of work and tool-building (and new opportunities for the same) have been observed in other academic communities (Ruhleder 1991, 1995; Weedman 1995). One person was there in the early days of the database *acedb*, and still contributed regularly, sending e mail about bugs and suggestions for graphics. Others constructed local tools, such as annotated gene lists (a project carried out part-time over the course of a year), using data from WCS. Yet another person, as mentioned above, planned to team up with a computer scientist to develop tools for data visualization. Many of our respondents could list tools (from techniques to compilations of targeted information, to analysis software) that they would have liked to see added or perfected. The difficulty is that there are no clear rewards for this kind of work, except for the contributions the tool makes to one's own work. The biologist working with the computer scientist doesn't get any "credit" for this within his own discipline (he anticipated having tenure by the time this project would begin). As one post-doc put it in a comment appropriate for both sides, ". . . there are a hundred things that are useful, but you don't get a Ph.D. for [them]" (Jay Emery, PD).

4.4.5. Addressing Third-order Issues. Third-order issues are a feature of complex communities. They may become easier to observe during times of flux because they resist local resolution. Novel technologies, situations, and concerns create immediate resource requirements and gaps in learning that can be addressed locally. Over time, however, the interactions of these first-order and second-order issues combine to raise broader questions which push the magnitude of a "solution" out of the local realm and into the wider community.

Electronic access to data via WCS, for instance, calls into question not only local resource allocations, but broader institutional alliances and patterns of contributions at the disciplinary level. The resolution of these issues or conflicts (if, indeed, they are resolved) may result in the creation of new subspecialties, new requirements for a discipline or profession, new criteria for the conduct and evaluation of work, and new reward structures. Resolutions or "readjustments" will not only take place overtly (i.e., though a petition to a campus computing committee, or a decision to reallocate travel funds to a lab computing fund). They may be played out on a political level by individuals with high stakes in maintaining stature or controlling resources, or they may be resolved serendipitously, even unconsciously. For instance, as mentioned above, questions of access to the WCS and the maintenance of an open, democratic structure within the community may become moot as other forms of access through the Internet become easier.

5. Double Binds: The Transcontextual Syndrome on the Net

Until now we have simply followed Bateson's typology for learning in categorizing infrastructural barriers and challenges. Bateson's ideas about levels of learning originated in communication theory and cybernetics; more than a taxonomy, they are an expression of set of dynamics:

> "Double bind theory is concerned with the experiential component in the genesis of tangles in the rules or premises of habit I . . assert that experienced breaches in the weave of contextual structure are in fact 'double binds' and must necessarily (if they contribute at all to the hierarchic processes of learning and adaptation) promote what I am calling transcontextual syndrome " (Bateson 1978, p. 276)

The formal statement of the problem is expressed as a logical one, following as we noted earlier Russell and Whitehead's theory of classification. In "The Logical Categories of Learning and Communication" (pp. 279–308), Bateson notes that a category error such as confusing the name of a class and a member of that class will create a logical paradox. In the world of pure logic, this appears as a fatal error, because such logical systems seem to exist outside of time and space. In the real world, particularly the behavioral world, however, people cope by working within multiple frameworks or "world views," maintained serially or in parallel

When messages are given at more than one level simultaneously, or an answer is simultaneously demanded at a higher level and negated on a lower one, there arises a logical paradox or "double bind," an instance of what Bateson terms the "transcontextual syndrome." While Bateson drew his examples from family contexts in the course of his work on schizophrenia, double binds occur in academic and business contexts as well. Middle managers in rapidly-changing environments, for instance, are frequently caught between the goals and expectations articulated by senior management and the actions of senior management with respect to budget allocation and performance evaluation (Mishra and Cameron 1991). Companies may formally promote efforts towards "reengineering" and "empowerment," yet offer no mechanisms for employees to participate in decision making, or they may sanction employees for not being active learners while refusing to acknowledge modes of learning and experimentation that fail to conform to very specific models (Ruhleder and Jordan, in progress). In the words of Bateson, "There may be incongruence or conflict between context and metacontext." (p. 245). Over a protracted period of time, with many such messages, schizophrenia may result, either literally or figuratively.[4]

People attempting to hook up to complex electronic information systems encounter a similar discontinuity between message types. The rhetoric surrounding the Internet makes "signing on" and "hooking up" sound remarkably straightforward. Furthermore, the benefits sound instantaneous and far-reaching. Why, then, do so many problems arise when members of the worm community try to take a similar step? Why are there so many disappointments with accessing information, and how may we understand these?

We identify several varieties of double binds arising across two levels or orders from what we call *infrastructural transcontextual syndrome:*

1. The gap between diverse contexts of usage,
2. The gap inherent in various computing-related discussions within the worm community itself, and
3. The gulf between "double levels of language" in design and use.

The gap between diverse contexts of usage. What is simple for one group is not for the other, so what appears to be a level one message to computer scientists posed a level two problem for users, creating a double bind. For instance, when asked about getting onto the system, designers of WCS might say, "Just throw up X Windows and FTP the file down." The tone of the message is clearly level one, a simple "recipe" for the UNIX-literate. For the relatively naive user, however, it requires them to move to a different contextual level and to figure out what type of a thing an "X Window" is, and what it means to FTP a file down. A level one instruction thus becomes a complex set of level two questions, closely related to the user's own level of expertise. These kinds of transcontextual difficulties will intensify as collaborative systems and groupware are developed for increasingly nonhomogenous user communities (Grudin 1991, 1994; Markussen 1994).

Another part of this type of double-bind is an infinite regress of barriers to finding out about complex elec-

[4] The child insists on seeing the literal level and ignoring context, or inappropriately seeing context literally. The often-noted poetry in schizophrenic language is a result of this refusal—good poets deliberately play with transcontextual double entendres Formally, this ignores or transgresses the gulf between message and metamessage

tronic information systems (Markussen 1995). If you don't know already, it's hard to know *how* to find out, and it isn't always clear how to abstract knowledge from one system to another. What is obvious to one person is not to another; the degrees of obviousness continue indefinitely, forming complex binds. For example, there is no single book that can tell you from scratch about computers or networked computing; the only way in is to switch contexts altogether and work more closely with those who already know, becoming a member of some community (Suchman and Trigg 1991, Suchman et al. 1986). This may account for the power of the participatory design model popular in Scandinavia, in which designers and users work together to the point of developing a shared context at all levels of interaction (Ehn 1988). It may simultaneously account for the difficulty of explaining or popularizing the model outside of Scandinavia, the working context of which differs greatly from the U.S. or other parts of Europe.[5]

The gap inherent in discussions within the worm community. Within the worm community itself there exists a level two–level three double bind. Just as level one statements can engender level two questions, so can level two discussions open up higher order issues. Discussions about package or platform choice become discussions about resource allocation, data interpretation, and network externalities. Take, again, the case of "FTP-ing a file down." Talk of learning about FTP, about alternatives such as gophers, etc., becomes questions of access across labs, of database maintenance and data reliability, and of norms and rewards within the community for contributions to the database.

These issues are particularly poignant ones for "older" members of a fairly new community, who recognize that technical choices and decisions made at the second level—evaluations of the options for responding to level one signals—have the ability to affect dramatically third order issues. In the worm community, the concerns involve changes in the composition of the community as "outsiders" join, and what this means for data interpretation and tool construction. The concerns also center around the multiple roles that research on the organism plays: "end in and of itself" vs. model organism for the Human Genome Initiatives. Tools aimed at second-level problems affect deeply the options open to the discipline when addressing third-order questions and setting broad conceptual directions.

Double levels of language in design and use. There may be double binds in those aspects of the system which are self-contradictory, between formal system properties and informal cultural practices. The language of design centers around technical capacity, while the language of use centers around effectiveness. Robinson (1991) notes that for systems that provide electronic support for computer-supported cooperative work, only those applications which simultaneously take into account both the formal, computational level and the informal, workplace/cultural level are successful. This problem is not unique to this domain. Gasser (1986), for instance, identifies a variety of "workarounds" developed to overcome the rigidity (and limitations) of a transaction processing system, while users of an insurance claims processing system developed an elaborate and informal set of procedures for articulating alternatives and inconsistencies (Gerson and Star 1986). Other examples abound.

While none of these studies identifies the problems/solutions as evidence of a double bind, each may be expressed in these terms. The "language" of the designer is focused on the technical representation of a particular set of data (i.e., customer records) and the efficiency of processing them to meet a particular goal (i.e., claims processing). The "language" of the user is focused on the need to mediate between conflicting viewpoints (i.e., doctors vs. representatives for large customer groups), and the need to develop effective workflows within their own workplaces. Orlikowski (1993) discusses more narrowly the conceptualization of software design methods and tools as languages and, together with Beath, examines the consequences of non-shared languages or organizational barriers to full participation of users (Beath and Orlikowski 1994).

This double-bind is also captured in the discussion of Mac vs. UNIX, and what it means in terms of a clash of cultures between biologists and computer scientists. On one level it is a discussion about operating systems, on

[5] Participatory Design has its own inherent difficulties (Markussen 1994, Nyce and Lowgren 1995)

another it is representative of two world views and sets of values with respect to the relationship between technology and work—the relationship between the tool and its user. In the case of WCS, designers focused on features of technical elegance and sophistication, such as constructing a mechanism for continuous *Gazette* updating, or fully exploiting hypertext possibilities. Yet constant information updating works against the biologists' informal mechanisms for information distribution, processing, and integration. And biologists were less interested in additional layers of complex hypertext linkages than in simple capabilities, like printouts of parts of the genetic map, which could be taken back to their lab benches, tacked up, pasted into a notebook, and easily annotated in the flow of work.

5.1. Summary and Recommendations for Addressing Double Binds

WCS—and the push for collaboratory development which set the stage for this and other projects (Lederberg and Uncapher 1988)—was driven by a desire not only to support collaborative scientific efforts, but to foster "ideal communities" of rich communication and seamless universal information access. WCS had the advantage of starting with a community in which many of those norms were already in operation, and whose small size made it relatively cohesive. It had a dedicated design team with knowledge of the target domain. It had an interested user population. Yet it never achieved its original goals and, while it does serve as a platform for communication and information access for some, others have found the barriers locally insurmountable, or the system itself superfluous.

When will WCS become infrastructure? The answer is, probably never in its original form, for the reasons outlined above. The development of the system and its integration into the community could not overcome the double binds that emerged within the context of system implementation and use. Nor could its development negate the impacts of other technologies such as gopher and Mosaic. Constructed largely as a series of "building blocks" available from other sources, it was easy enough for those building blocks to assemble or duplicate themselves elsewhere. But WCS and other systems development efforts based on this model of collaboratory *can* benefit from some of the lessons learned or newly illustrated through our analysis. And organizations interested in developing large-scale information and communication infrastructures (whether formal business organizations or loosely-coupled academic communities) can become aware of the efforts required on their part to meet developers halfway. Having identified different instances of double-binds that predicated the failure of WCS, we are left with the need to suggest positive action; we offer two recommendations below for addressing double binds.

5.1.1. The Role of Multidisciplinary Development Teams. One of the key difficulties with double binds is recognizing them in the first place: individuals involved in a situation may not be able to identify instances of this transcontextual syndrome. The other key difficulty, once a double bind is identified, is to articulate it such that the other party will recognize it as a problem. Dynamics of power and authority are clearly important here. In family settings, a parent might reject affectionate behavior on the part of the child, then, when the child withdraws, accuse the child of not loving the parent. The child may not always have capacity for analyzing and correcting this inconsistency, just as employees may not really have the power to address problems in a business environment that overtly empowers them. Managers may even subtly sanction the "wrong" kind of empowered behavior. Users are often given computer-based tools that are either cumbersome or ill-explained to them; when they fail to use them, they are labeled as being "resistant" to technology (Markus 1983, Markus and Bjørn-Andersen 1987, Forsythe 1992).[6]

A computing-related analogue would be the denial on the part of developers or system administrators that technical difficulties really mask higher-order conceptual problems centered around work practices and community standards, and a failure on the part of users to recognize the complexity of their work domains, their hidden assumptions, and the various motivations of the stakeholders involved. If we expect designers to learn about the formal and informal aspects of the user do-

[6] There is an analogy here with medicine, viz., studies of "patient" compliance which overlook the infrastructural and political features of medicine itself. See, e g., Strauss 1979, Strauss et al 1985.

main, to learn to "speak their language," we must ask users to meet designers halfway by learning their language and developing an understanding of the design domain. If designers are at fault for assuming that all user requirements can be formally captured and codified, users are often equally at fault for expecting "magic bullets"—technical systems that will solve social or organizational problems.

The fault may really lie in neither camp. Often miscommunication resulting in the double-binds of language, and the context within which the *process* of design/use occurs are responsible. The emergence of multi-disciplinary development teams may help to alleviate aspects of the transcontextual syndrome identified above, with ethnographers helping users and designers bridge the contextual divide. "You can FTP that from such-and-such a site" might well give way to "I can give you the FTP address, but the kind of data you'll get won't be detailed enough for what you want to do with it." By sharing an understanding of both the formal, computational level (traditionally the domain of the computer programming and systems analyst) and the informal level of workplace culture, double binds may be more easily identified as all members of the team learn to correctly identify the various orders or levels to which a message might belong. This sharing, however, requires institutional contexts that support and even reward this kind of collaboration.

5.1.2. The Nature of Technical User Education. Many elements of the "computing infrastructures" emerging within the academic and business communities are not custom made. They consist of locally-developed applications, off-the-shelf packages or tailored applications, local area networks and the Internet, commercial on-line services, and "shareware" such as Mosaic and Netscape. They vary greatly in terms of stability, maintainability, interoperability, and access to support. Yet in order to carry out their work effectively in increasingly computer-based environments, individuals must be able to negotiate complex configurations of technical resources. Pentland's (in press) analysis of software help lines attests to this complexity: "Software support is an activity that occurs on the 'bleeding edge' of technology, on the boundary between the known and the unknown." (p. 1 of ms) Support technicians and customers are often speaking from two disparate viewpoints, and successful support means recognizing and juggling this reality (Heylighen 1991). The emergence of local "tailors" (Trigg and Bødker 1994) and "technology mediators" (Okamura, et al. 1994) may provide a bridge between relatively generic technologies and their local interpretation and application.

Individuals are being told that they must adapt to new technologies and become technically literate, yet the type of training and support offered to them rarely gives them the basis necessary to evolve along with the infrastructure. Training sessions, on-line tutorials, and user manuals focus on a set of skills limited to particular applications, and occur outside of the context of actual work (Bjork 1994). Computer support centers may assist individuals in situ, but tend to be reactive, imparting one solution at a time, without any contextual connection to the kinds of technical problems the user has had before. To apply Bateson's framework, they are aimed at giving people the skills to address first order technical issues, though broader issues—such as whether the implementation of a particular groupware technology is consistent with local, career, or global strategic direction—may require second- or third-order conceptual skills.

Frameworks for various levels of "computer literacy" already exist within the computer science and education communities. What are missing are institutional mechanisms—whether the "institution" is a business enterprise, a university, or a scientific discipline—to support individuals in two ways. First, they do still need to teach specific skills, but they need to place these skills within a technical context that enables users to apply them to the next application, and the next. Second, they need to assist users in developing and maintaining the kind of computer literacy that will allow them to understand and address second- and third-order issues, especially as they unfold over time—a kind of learning that occurs through on-going dialogue and experimentation. That literacy must thus be coupled with an understanding of emerging work practices (locally and more broadly within their organizations). Finally, organizations also need to develop mechanisms for legitimating and rewarding the work of local tailors and mediators.

These institutional mechanisms can be, in part, consciously constructed. But in order for them to grow dynamically along with emergent user expertise and an emergent base of computing technologies, they must be predicated on the notion that organizations function as complex communities, and that learning takes place within local communities of practice (Lave and Wenger 1992, Star 1995). The creation and use of discrete technologies must occur within a broader context which is constantly reified by participants within and across the various communities of practice which define a particular organization. The success of systems developed to support their work is predicated on the creation of shared objects and practices, boundary objects, and infrastructures (Star and Griesemer 1989, Star 1989b). For instance, use of WCS was and continues to be predicated on the complex interaction between a variety of small and large communities: the WCS development team, a nonhomogeneous target population (the worm community), local systems support groups, and remote data collection and distribution centers. Each of these constitute extant, partially-overlapping communities of practice. Discontinuities in these interactions, unequal participation, and the emergence and continued rise of competing technologies have contributed to the inability of WCS to emerge as boundary object or fully to submerge as infrastructure.

6. Organizational Environment: Communities and Large-scale Infrastructure

Using the analysis put forward in the previous section, we would like to understand the nature of the claims about community and the net as examples of the complex emergence of infrastructure. We see a number of ways in which the merging of medium and message in the talk about scientific electronic communities is problematic, in addition to the double bind/transcontextual syndrome issues. Scientists do not "live on the net." They do make increasingly heavy use of it; participation is increasingly mandatory for professional advancement or even participation, with a rapidly changing set of information resources radically altering the landscape of information "user" and "provider" (Klobas 1994); and the density of interconnections and infrastructural development is proceeding at a dizzying rate. That development is uneven; is an interesting mixture of local politics and practices, on-line and off-line interactions, and filled with constantly shifting boundaries between lines of work, cohorts and career stages, physical, virtual and material culture, and increasingly urgent and interesting problems of scale.

The multiple meanings of WCS for different groups and individuals are useful as exemplars for understanding the challenges posed by "the net." From one perspective, the WCS fits well the cognitive map of the scientist with respect to information: links between disparate pieces, graphical representations, layers of detail, etc. Yet relatively few worm biologists have "signed on" to WCS, even as the community itself is growing rapidly. The seeming paradox of why our respondents chose to use Gopher, Mosaic and other simpler, public-access systems rather than WCS involves a kind of double bind at larger scale.

To take on board the custom-designed, powerful WCS (with its convenient interface) is to suffer inconvenience at the intersection of work habits, computer use, and lab resources. Its acquisition disrupts resource allocation patterns: on-going use and support requires an investment in changes of habit and infrastructure. The World-Wide Web, on the other hand, can be accessed from a broad variety of terminals and connections, and Internet computer support is readily available at most academic institutions and through relatively inexpensive commercial services.

Yet even within the larger context of infrastructure, there are other ways in which WCS serves its community less well than alternate emerging infrastructures. Science is an integrative and permutable domain, and requires a complimentary infrastructure (Ruhleder and King 1991). The construction of the WCS, while it integrates a large number of materials, does so in a constricted fashion. Lab notebooks, by way of contrast, are extremely open and integrative documents (Gorry et al. 1991). At the same time, computing infrastructures, including gophers, FTP sites, etc., while still at a very primitive level, fit more closely with this integrative model than relatively closed systems such as the WCS, and these infrastructures are growing at a phenomenal rate. For these reasons, and despite frustrations over the lack of indexing and search capabilities, use of Gopher and Mosaic within the *c.elegans* world abounds.

7. Conclusion

Can an organizational support system be developed that allows people to coordinate large-scale efforts, provide navigational aids for newcomers, yet still retain the feeling of an informal, close-knit community or cohesive organizational culture? If structure is not incorporated a priori, then does it emerge, and how? Just as the WCS was intended to bridge geographic and disciplinary boundaries within the worm community, groupware and related technologies are being constructed as technical infrastructures to support members of an organization in bridging physical, temporal, and functional boundaries.

Experience with groupware suggests that highly structured applications for collaboration will fail to become integrated into local work practices (Ruhleder, Jordan, and Elmes in progress). Rather, experimentation over time results in the emergence of a complex constellation of locally-tailored applications and repositories, combined with pockets of local knowledge and expertise. They begin to interweave themselves with elements of the formal infrastructure to create a unique and evolving hybrid. This evolution is facilitated by those elements of the formal structure which support the redefinition of local roles and the emergence of communities of practice around the intersection of specific technologies and types of problems. These observations suggest streams of research that continue to explore how infrastructures evolve over time, and how "formal," planned structure melds with or gives way to "informal," locallyemergent structure.

The competing requirements of openness and malleability, coupled with structure and navigability, create a fascinating design challenge—even a new science. The emergence of an infrastructure—the "when" of complete transparency—is thus an "organic" one, evolving in response to the community evolution and adoption of infrastructure as natural, involving new forms and conventions that we cannot yet imagine. At the same time, it is highly challenging technically, requiring new forms of computability that are both socially situated and abstract enough to travel across time and space (Eveland and Bikson 1987, Feldman 1987). Goguen (1994) and Jirotka and Goguen (1994) recently referred to these as "abstract situated data types" for requirements analysis, and notes that requirements engineering in this view in fact *becomes* "the reconciliation of technical and social issues."

In the end it seems that organizational change and the resolution into infrastructure are usually very slow processes. Local and large-scale rhythms of change are often mismatched, and what it takes to really make anything like a national or global information space is at the very cutting edge of both social and information science. The mixture of close-in, long-term understanding gained by ethnography and the complex indexing, programming and transmission tasks afforded by computer science meet here, breaking traditional disciplinary boundaries and reflecting the very nature of the problem: when is an ecology of infrastructure?[7]

[7] WCS was partially funded by the NSF under grants IRI-90-15047, IRI-92-57252, and BIR-93-19844, and currently is housed within the Community Systems Laboratory (CSL), affiliated with the Graduate School of Library and Information Science and the National Center for Supercomputing Applications at the University of Illinois, Urbana-Champaign Additional support was provided by the University of Arizona and the University of Illinois. Co-PIs Bruce Schatz and Sam Ward, and developers Terry Friedman and Ed Grossman were extremely generous with their time, comments, and access to data and meetings; we also thank our anonymous respondents for their time and insight. Earlier versions of this paper were presented at the *Conference on Computing in the Social Sciences*, 1993, *CSCW* 1994, and *ASIS* 1995 Ruhleder thanks Michael Elmes for an interesting discussion on double binds in organizations, and thanks Sam Politz and the graduate students in his worm lab at Worcester Polytechnic Institute for teaching her how to hook and breed worms, and how to run a gel. Marc Berg, Geof Bowker, Nick Burbules, Tom Jewett, Alaina Kanfer, Rob Kling, Jim Nyce, Kevin Powell and Stefan Timmermans provided helpful insights and comments Star's work was also supported by the Program in Culture Values and Ethics and the Advanced Information Technologies Group, University of Illinois; the Institute for Research on Learning, Palo Alto, and by an NSF Professional Development Grant The authors would like to thank John Garrett of CNRI; Tone Bratteteig, Pål Søregaard, Eevi Beck, Kari Thoresen, Ole Hanseth, Eric Monteiro and the Internet Project working group at the Institute for Informatics, University of Olso, Yrjö Engestrom, Chuck Goodwin and Dick Boland for discussions and challenges about the concept of infrastructure The dimensions of infrastructure outlined were partly developed in an email dialogue between Star and Garrett; we gratefully acknowledge his help and insight Pauline Cochrane and the students in her information retrieval seminar provided very helpful comments, we acknowledge their help and that of Laura Shoman

References

Anderson, William L and William T. Crocca, "Engineering Practice and Codevelopment of Product Prototypes," *Comm. ACM*, 36 (1993), 49–56

Barlow, John Perry, "Is There a There in Cyberspace?" *Utne Reader*, 68 (March–April 1995), 53–56

Bateson, Gregory, *Steps to an Ecology of Mind*, Ballantine Books, New York, 1978.

Beath, Cynthia and Wanda J. Orlikowski, "The Contradictory Structure of Systems Development Methodologies: Deconstructing the IS-User Relationship in Information Engineering," *Information Systems Res.*, 5, 4 (1994), 350–377

Becker, Howard S., *Art Worlds*, University of California Press, Berkeley, 1982.

Bødker, Susanne, *Through the Interface*, Erlbaum, Hillsdale, NJ, 1991

Bowker, Geoffrey, "Information Mythology and Infrastructure," in Lisa Bud-Frierman (Ed.), *Information Acumen: The Understanding and Use of Knowledge in Modern Business*, Routledge, London, 1994, 231–247.

——, "The Universal Language and the Distributed Passage Point: The Case of Cybernetics," *Social Studies of Sci.*, 23 (1993), 107–127.

Brown, John Seely and Paul Duguid, "Borderline Issues. Social and Material Aspects of Design," *Human-Computer Interaction*, 9 (1994), 3–36

Bucciarelli, Louis L., *Designing Engineers*, MIT Press, Cambridge, MA, 1994.

David, Paul, "Clio and the Economics of QWERTY," *American Economic Rev.*, 75 (1985), 332–337.

Davies, Lynda and Geoff Mitchell, "The Dual Nature of the Impact of IT on Organizational Transformations," in R. Baskerville, O. Ngwenyama, S. Smithson, and J. DeGross (Eds.), *Transforming Organizations with Information Technology*, North Holland, Amsterdam, 1994, 243–261.

Ehn, Pelle, *Work-Oriented Design of Computer Artifacts*, Lawrence Erlbaum, Hillside, NJ, 1988

Ellis, C. A., S. J. Gibbs, and G. L. Rein, "Groupware: Some Issues and Experiences," *Comm. ACM*, 34 (1991), 38–58

Engestrom, Yrjö, "When Is a Tool? Multiple Meanings of Artifacts in Human Activity," in his *Learning, Working and Imagining*, Orienta-Konsultit Oy, Helsinki, 1990.

Eveland, J. D. and Tora Bikson, "Evolving Electronic Communication Networks: An Empirical Assessment," *Office. Technology and People*, 3 (1987), 103–128.

Feldman, Martha, "Constraints on Communication and Electronic Messaging," *Office: Technology and People*, 3 (1987)

Forsythe, Diana, "Blaming the User in Medical Informatics. The Cultural Nature of Scientific Practice," in *Knowledge and Society: The Anthropology of Science and Technology*, Vol. 9, David Hess and Linda Layne (Eds.), JAI Press. Greenwich, CT, 1992, 95–111

Gasser, Les, "The Integration of Computing and Routine Work," *ACM Trans. Office Information Systems*, 4 (1986), 205–225.

Gerson, E. M. and Susan Leigh Star, "Analyzing Due Process in the Workplace," *ACM Trans. Office Information Systems*, 4 (1986), 257–270

Gorry, G. Anthony, Kevin B. Long, Andrew M. Burger, Cynthia P. Jung, and Barry D. Meyer, "The Virtual Notebook System^{TM} An Architecture for Collaborative Work," *J. Organizational Computing*, 1 (1991), 223–250

Grudin, Jonathan, "Obstacles to User Involvement in Software Product Development, with Implications for CSCW," *International J. Man-Machine Studies*, 34 (1991), 435–452.

——, "Groupware and Social Dynamics. Eight Challenges for Developers," *Comm. ACM*, 37, 1 (1994), 92–105

Goguen, Joseph, "Requirements Engineering as the Reconciliation of Technical and Social Issues," in M. Jirotka and J. Goguen (Eds.), *Requirements Engineering: Social and Technical Issues*, Academic Press, NY, 1994

Hewitt, Carl, "Offices Are Open Systems," *ACM Trans. Office Information Systems*, 4 (1986), 271–287.

Heylighen, Francis, "Design of a Hypermedia Interface Translating Between Associative and Formal Representations," *International J. Man-Machine Studies*, 35 (1991), 491–515

Hughes, Thomas P., *Networks of Power: Electrification in Western Society*, 1880–1930, Johns Hopkins University Press, Baltimore, MD, 1983

——, "The Evolution of Large Technological Systems," in Wiebe E. Bijker, Thomas P. Hughes, and Trevor Pinch (Eds.), *The Social Construction of Technological Systems*, MIT Press, Cambridge, 1989, 51–82

Jewett, Tom and Rob Kling, "The Dynamics of Computerization in a Social Science Research Team: A Case Study of Infrastructure, Strategies, and Skills," *Social Sci. Computer Rev.*, 9 (1991), 246–275

Jirotka, Marine and Joseph Goguen, *Requirements Engineering: Social and Technical Issues*, Academic Press, New York, 1994.

Kling, R. and W. Scacchi, "The Web of Computing: Computing Technology as Social Organization," *Advances in Computers*, 21 (1982), 3–78

Klobas, Jane E., "Networked Information Resources: Electronic Opportunities for Users and Librarians," *Information Technology and People*, 7 (1994), 5–18

Korpela, Eija, "Path to Notes: A Networked Company Choosing Its Information Systems Solution," in R. Baskerville, O. Ngwenyama, S. Smithson, and J. DeGross (Eds.), *Transforming Organizations with Information Technology*, North Holland, Amsterdam, 1994, 219–242

Kraemer, Kenneth L. and John L. King, *Computers and Local Government*, Praeger, New York, 1977

Latour, Bruno, *We Have Never Been Modern*, Translated from the French by Catherine Porter, Harvard University Press, Cambridge, MA, 1993.

Lave, Jean and Etienne Wenger, *Situated Learning: Legitimate Peripheral Participation*, Cambridge University Press, Cambridge, 1992.

Lederberg, J. and K. Uncapher, "Towards a National Collaboratory," in *Report of an Invitational Workshop at the Rockefeller University*, National Science Foundation, Washington, DC, 1988

Malone, Tom and Gary Olson (Eds.), *Coordination Theory and Communication Technology*, Erlbaum, Hillsdale, NJ, in press.

Markus, M. Lynne, "Power, Politics, and MIS Implementation," *Comm. ACM*, 26, 6 (1983), 430–444

—— and Niels Bjørn-Andersen, "Power over Users. Its Exercise by System Professionals," *Comm. ACM*, 30, 6 (1987), 498–504.

Markussen, Randi, "Dilemmas in Cooperative Design," *PDC '94. Proc Participatory Design Conf.*, Computer Professionals for Social Responsibility, Palo Alto, CA, 1994, 59–66.

——, "Constructing Easiness Historical Perspectives on Work, Computerization, and Women," in Susan Leigh Star (Ed.), *The Cultures of Computing*, Basil Blackwell, Oxford, UK, 1995

Marx, Karl, *Capital Vol.* 1. Lawrence and Wishart, London, 1970

Mishra, Aneil K. and Kim S. Cameron, "Double Binds in Organizations· Archetypes, Consequences, and Solutions From the US Auto Industry," Presentation at the Academy of Management Annual Meeting, August 13, 1991.

Monteiro, Eric, Ole Hanseth and Morten Hatling, "Developing Information Infrastructure· Standardization vs. Flexibility," Working Paper 18 in *Science, Technology and Society*, University of Trondheim, Norway, 1994.

Nyce, James and Jonas Löwgren, "Toward Foundational Analysis in Human-Computer Interaction," in Peter J. Thomas (Ed.), *The Social and Interactional Dimensions of Human-Computer Interfaces*, Cambridge University Press, Cambridge, UK, 1995

Okamura, Kazuo, Masayo Fujimoto, Wanda J. Orlikowski, and JoAnne Yates, "Helping CSCW Applications Succeed: The Role of Mediators in the Context of Use," in *Proc Conf. Computer Supported Cooperative Work*, ACM Press, Chapel Hill, NC, October 22-26, 1994.

Orlikowski, Wanda, "Integrated Information Environment or Matrix of Control? The Contradictory Implications of Information Technology," *Accounting, Management and Information Technology*, 1 (1991), 9–42.

——, "CASE Tools as Organizational Change: Investigating Incremental and Radical Changes in Systems Development (Computer-Aided Software Engineering)," *MIS Quarterly*, 17 (1993), 309–340

Pentland, Brian, "Bleeding Edge Epistemology: Practical Problem Solving in Software Support Hot Lines," in Stephen Barley and Julia Orr (Eds.), *Between Technology and Society: Technical Work in the Emerging Economy*, Submitted to ILR Press, in press

Pool, Robert, "Beyond Databases and Email," *Science* 261 (13 August 1993), 841–843.

Robinson, Mike, "Double-level Languages and Co-operative Working," *AI and Society* 5 (1991), 34–60

Ruhleder, Karen, "Information Technologies as Instruments of Transformation: Changes to Work Processes and Work Structure Effected by the Computerization of Classical Scholarship," Ph D dissertation, Dept. of Information and Computer Science, University of California, Irvine, CA, 1991.

——, "Reconstructing Artifacts, Reconstructing Work: From Textual Edition to On-line Databank," *Sci. Technology and Human Values*, 20 (1995), 39–64.

——, Brigitte Jordan, and Michael Elmes. "Learning in the 'New Organization': Integrating Collaborative Technologies and Team-Based Work," *Accepted to the Academy of Management Annual Meeting*, Cincinnati, OH, August 9-12, 1996.

—— and John Leslie King, "Computer Support for Work Across Space, Time and Social Worlds," *J. Organizational Computing*, 1, 4 (1991), 341–356.

Schatz, Bruce, "Building an Electronic Community System," *J. Management Information Systems*, 8 (1991), 87–107

Schmidt, Kjeld and Liam Bannon, "Taking CSCW Seriously Supporting Articulation Work," *Computer Supported Cooperative Work (CSCW): An International J.*, 1 (1992), 7–41

Simmel, Georg, *The Sociology of George Simmel*, Kurt Wolff (Ed.), Free Press, Glencoe, IL, 1908, 1950.

Star, Susan Leigh, *Regions of the Mind: Brain Research and the Quest for Scientific Certainty*, Stanford University Press, Stanford, CA, 1989a

——, "The Structure of Ill-Structured Solutions. Heterogeneous Problem-Solving, Boundary Objects and Distributed Artificial Intelligence," in M Huhns and L Gasser (Eds), *Distributed Artificial Intelligence* 2, Morgan Kauffmann, Menlo Park, NJ, 1989b, 37–54

——, "Power, Technologies and the Phenomenology of Conventions: On Being Allergic to Onions," in John Law (Ed), *A Sociology of Monsters Essays on Power, Technology and Domination*, Routledge, London, 1991a

——, "Organizational Aspects of Implementing a Large-Scale Information System in a Scientific Community," Technical Report, Community Systems Laboratory, University of Arizona, Tucson, November 1991b

——, "Misplaced Concretism and Concrete Situations: Feminism, Method and Information Technology," Working Paper 11, Gender-Nature-Culture Feminist Research Network Series, Odense University, Odense, Denmark, 1994

——, "From Hestia to Home Page· Feminism and the Concept of Home in Cyberspace," in Nina Lykke and Rosi Braidotti (Eds.), *Between Monsters, Goddesses and Cyborgs: Feminist Confrontations with Science, Medicine and Cyberspace*, ZED-Books, London, in press

——, Ed., *The Cultures of Computing*, Blackwell, Oxford, 1995

Strauss, Anselm, *Qualitative Methods for Social Scientists*, Cambridge University Press, Cambridge, UK, 1986.

——, Ed , *Where Medicine Fails*, Transaction Books, NJ, 1979.

——, S Fagerhaugh, B Suczek, and C. Wiener, *Social Organization of Medical Work*, University of Chicago Press, Chicago, 1985.

Suchman, L and R Trigg, "Understanding Practice· Video as a Medium for Reflection and Design," in J. Greenbaum and M Kyng, (Eds), *Design at Work*, Lawrence Erlbaum, London, 1991, 65–89.

——, Randy Trigg, and F Halasz, "Supporting Collaboration in NoteCards," in D Peterson (Ed.), *Proc. Conf. Computer-Supported Cooperative Work* (*CSCW-86*), Austin, TX; ACM, New York, 1986, 1–10

Tönnies, Ferdinand, *Community & Society* (*Gemeinschaft und Gesellschaft*), translated and edited by Charles P. Loomis; Michigan State University Press, East Lansing, MI, 1957

Trigg, Randall and Susanne Bødker, "From Implementation to Design. Tailoring and the Emergence of Systematization in CSCW," in *Proc ACM* 1994 *Conf Computer-Supported Cooperative Work*, ACM Press, New York, 1994, 45–54.

Weedman, Judith, "Incentive Structures and Multidisciplinary Research· The Sequoia 2000 Project," paper presented at the American Society for Information Science, Chicago, IL, October, 1995.

Yates, JoAnne, *Control Through Communication The Rise of System in American Management*, Johns Hopkins University Press, Baltimore, MD, 1989.

Accepted by JoAnne Yates and John Van Maanen acting as Special Editors