REACH, BRACKET, AND THE LIMITS OF RATIONALIZED COORDINATION: SOME CHALLENGES FOR CSCW

Elihu M. Gerson
emg@tremontresearch.org
TREMONT RESEARCH INSTITUTE
458 Twenty-Ninth Street
San Francisco, CA 94131
415-285-7837 FAX 415-648-7660

Revised September, 2004

Prepared for a volume on "Resources, Co-Evolution, and Artifacts: Theory in CSCW", edited by Mark S. Ackerman, Christine Halverson, Tomas Erickson, and Wendy A. Kellogg. I am grateful to Les Gasser, Arnold Goldfein, Christine Halverson, Carl Hewitt, Walter Scacchi, Kjeld Schmidt, and Carla Simone for helpful conversations over many years; to M. Sue Gerson, Christine Halverson and an anonymous referee for helpful suggestions: to the editors for extraordinary patience; and to M. Sue Gerson for continuing support.

INTRODUCTION

There has been a long-term shift in the way computing arrangements support other kinds of work. The first generation of computing technology speeded up and automated the performance of standardized repetitive tasks, such as payroll preparation. Later generations concentrated on supporting and augmenting varying but routine tasks: desktop systems running "productivity" applications such as word processors and spreadsheets were typical applications.[1] More recently, the development of the World Wide Web, broadband communications, wireless technology, standardized markup languages, and other innovations have created many new contingencies and possibilities. Computing has become part of the day-to-day and even minute-to-minute flow of activities, and significant computational capacity is becoming ubiquitous. As a result, attention has increasingly shifted to understanding and supporting communication and cooperation among people, rather than automating specific tasks. The focus is thus now on helping people deal with one another conveniently, and performing loosely-defined tasks well enough. This requires allowing for unpredictable contexts and changing needs.

These trends have two major implications for the relationship of computing and work. The first implication is *hyper-distribution*. For many years, the movement of computing capacity from centralized installations in large organizations to smaller installations in smaller organizations changed the basic relationships between computing and work incrementally, and in predictable ways. Recently however, we have crossed a threshold in the ways information systems support work. Joint activities are now easily conducted across organizational boundary lines. Geographical barriers to coordination and cooperation are also decreasing quickly. While face-to-face interaction is still needed for many purposes, steady progress in the quality and cost of telecommunications is steadily chipping away at the need for meetings in which people are physically present together.

The second implication of these trends in computing is *hyper-accessibility*. An increasing number of objects and tasks are becoming capable (at least in principle) of finding and communicating with one another across distances and organizational boundaries. For example, it has already become routine for automobiles to report their locations, a capacity which gladdens the hearts of car rental firms, law enforcement agencies, and perhaps the parents of teen-agers. The emergence of Bluetooth and RFID technology raises the possibility that we will shortly see extensions of this trend, as individual objects begin to interact with the places and organizations which contain them. While this trend certainly provides opportunities for improved coordination, new services, and many other benefits, it also entails many new or increased difficulties in managing privacy, security, accountability and reliability.

"Hyper-distribution and hyper-accessibility" is an awkward phrase. We need a convenient term to refer to the distribution of tasks across organizational, spatial, and temporal boundaries. I'll call this the "reach" of a task. Increased reach is having many far-reaching consequences for the organization of work. It is removing or reducing many kinds of constraint, and thus makes new arrangements feasible. At the same time, it is creating new problems that require new kinds of constraint. For example, a mobile telephone with a built-in camera and Internet access is a wonderful thing when I'm in the store and I want to ask my wife which model widget we should buy, but I don't want someone following me into the public restroom with one.

As a result of these trends, many activities that were conducted on a person-to-person basis no longer have the immediacy that lets people see and hear what other participants are doing. One example is the loss of emotional context when using email and instant messaging

rather than face-to-face or even telephone communication. Much important information about mood and temper is lost. More generally, the trend implies a loss of context for many kinds of interaction. Increasingly, interacting parties don't have good knowledge of one another's history and circumstances. This limits the trust which parties can place in one another. Coordinating tasks in these circumstances means that formal mechanisms such as passwords, ID cards, and credit checks are needed to assure participants of reliability and safety. Interaction among strangers is thus becoming explicitly controlled, formal, and mediated by a variety of devices and procedures. Increasing reach also means that work is becoming more intensively computer-supported. CSCW research is thus becoming relevant, in principle, to almost every kind of work. This increased relevance requires careful thinking about the new ways in which work is performed, how tasks interact, and how work interacts with information systems.

COORDINATION MECHANISMS

Increasing reach implies that the problem of coordinating tasks and people (never easy) becomes markedly more complex. Resources and results must be brought together at appropriate times and places if the work is to succeed. Of course, there are many ways of addressing this question; it is the defining problem for research in management science. Two classical treatments are Lawrence and Lorsch 1967, and Thompson 1967. Malone and Crowston (1994) argue for an emerging research area focused explicitly on coordination, which they define as the management of dependencies among activities. Here, I want to use a different definition of coordination, one focused on the ways in which the parts of an activity function (or fail to function) together. In particular, I am concerned with the implications of increasing reach for the co-functioning of tasks. The question for CSCW is: what role does computing technology play in this process?

The very general and powerful notion of "coordination mechanism" proposed by Schmidt and Simone in 1996, helps us specify the issues involved in a useful way:

> A *coordination mechanism* is a specific organizational construct, consisting of a *coordinative protocol* imprinted upon a *distinct artifact*, which, in the context of a certain cooperative work arrangement *stipulates* and *mediates* the articulation of cooperative work so as to *reduce the complexity of articulation*

> *work* of that arrangement. (Schmidt and Simone 1996: 180; emphasis in original)

Schmidt and Simone define coordination mechanisms in term of distinct artifacts. The prototypical artifacts of this kind are business forms such as bills of lading or restaurant checks. An alternative approach, adopted here, is to think of coordinating mechanisms as tasks dedicated to orchestrating the work of other tasks. Often, this work makes use of objects which serve as a focus of the interaction-- typically, business forms which carry information from one task to another and back again. Our primary concern however, is with the work that meshes two or more activities. Hence, it's useful to think of a coordinating mechanism as consisting of both an artifact and the work of using it-- e.g., filling out and interpreting forms "correctly". For example, a restaurant check serves to coordinate the interaction of server with patron, server with kitchen staff, and patron with cashier. This works, of course, only if everyone uses the check in the way it was intended. So the work of preparing and using the device correctly is a part of the mechanism, which is this best considered as specialized kind of task.

The idea of coordinating mechanism is motivated by the idea of articulation work developed by Anselm Strauss. Strauss used the notion of articulation work in two different senses (e.g., Strauss 1987). On the one hand, articulation work is about making sure all the various resources needed to accomplish something are in place and functioning where and when they're needed *in the local situation*. This means bringing together everything needed to accomplish a task at a particular time and place, including all the administrative and support functions such as janitorial services, food service, equipment maintenance, and covering for staff out sick or on vacation. The concern and emphasis in this sense are on particular situations rather than classes of activity.

In its second sense, articulation work means "putting together tasks, task sequences, task clusters-- even aligning larger units such as lines of work and subprojects-- in the service of work flow" (Strauss 1991: 100). In this second sense, the focus is not so much on the specifics of work in a particular local situation, as it is on making sure that different *kinds* of activity function together well. The two senses, of course, overlap heavily-- especially when all the tasks are part of the same organization and are carried out in the same place.

The notion of articulation has been used in the CSCW literature most often in the second sense, which I'll call *metawork*-- the work of making work go well. I'll keep the term

"articulation" (or "local articulation") to refer to the first of Strauss' senses. We can always distinguish articulation work from metawork because articulation work is about a particular situation. So, for example, the supplies cabinet in every office has to be re-stocked from time to time. Restocking as articulation work consists of putting needed supplies in a particular cabinet on a particular date. Metawork, by contrast, consists of specifying what restocking consists of. It's the distinction between specifying what goes on a pre-printed shopping list, and checking off the items on a copy of the list for today's shopping trip.

The distinction between the two senses isn't very important when we're focused on single organizations, as Strauss did in his 1987 article. There, he explicitly set aside the problems associated with analyzing articulation of single projects across multiple organizations, multiple worlds, and multiple sites. The importance of increasing reach however, means that we must make the distinction clear and understand its implications. The work of specifying the work to be done is one thing; the work of ensuring performance in specific circumstances is another.

The analysis of coordination mechanisms then, is primarily about metawork-- although, as we shall see, articulation has an important part to play as well. A coordination mechanism is something that activities use to orient their courses vis-à-vis one another. It is thus part of the work-flow in two different activities simultaneously, and serves to make them contingent upon one another. A coordinating mechanism may be as simple as a job ticket, on which the progress of a joint task is marked for all participants, or it may be a complex activity in its own right.

"Coordination mechanism" is another awkward term, especially since I am going to be discussing coordination in other senses below. Hence, I will adopt the term "bracket" as a synonym for "coordination mechanism". The significant things about a bracket (or bracketing as an activity) are (1) it connects two things together, and makes them part of a larger system of dependencies; (2) it does so in specific ways; and (3) it also holds them apart and keeps them distinct. I mean all of these senses to apply when I write here of brackets. I am trying to call to mind, not the specialized punctuation marks which often signal the presence of metawork (thus), but the pieces of metal which fasten and support shelves to a bookcase or a wall.

REDUCING THE COMPLEXITY OF METAWORK AND ARTICULATION: VARIETIES OF RATIONALIZATION

In the view of Schmidt and Simone, the function of a coordinating mechanism is to reduce the complexity of metawork and articulation work-- to simplify the ways in which the parts of a system interact. An effective mechanism thus helps to *rationalize* the interaction of the coordinated tasks. I use the term rationalization here in the economist's sense of doing more with the same resources, or the same work with less resources. This seemingly simple idea turns out to be fairly complex, because there are three kinds of rationalization, and sometimes they trade off against one another.

One kind of rationalization is *segregating* rationalization. This makes things independent of one another, removing the connections or contingencies among them wherever possible. Complex tasks are broken into multiple independent tasks. Similar things are grouped, and dissimilar things are segregated. Things are treated as unrelated members of a set, rather than as parts of a whole. Finally, relationships among things are treated as unary properties of their constituent units.

The second kind of rationalization is *standardizing* rationalization. This makes the connections and relationships among things uniform. Standardization substitutes repetitive similarities for unique or diverse activities, materials, tools, and situations. In doing so, it makes dealing with them cheaper, more convenient, and more reliable.

The third kind of rationalization is *coordinating* rationalization. This works by fine-tuning and refining relationships so that they are particularly well-suited to their situation. Within a given activity, tasks are made more responsive to one another by removing everything that does not contribute directly to smooth functioning, and by strengthening everything that does. Frequently (but not necessarily) it means taking advantage of specialized local circumstances or knowledge in order to create specialized local short-cuts.

All three ways of rationalizing work by simplifying relationships: by removing them, homogenizing them, or refining and specializing them. All three ways of rationalizing make activities more efficient; more gets done with the same resources, or it takes fewer resources to accomplish a given task.

Sometimes, the different rationalization processes work together or reinforce one another. For example, increasing standardization of raw materials and parts makes it far more practical to segregate manufacturing steps which assemble the parts into finished

products. This was precisely the advantage of the American System of manufactures, introduced in the nineteenth century (Hounshell 1984). In the older system, parts were hand-fitted together by skilled craftsmen. In the American System, parts were manufactured in high volume to close tolerances, so that they were interchangeable. As a result, assembly of the finished product required much less skill.

Standardization of materials and parts means that different manufacturing steps can be segregated temporally, spatially, and organizationally. The performance of each task can in turn be standardized, e.g., via time and motion engineering (cf. e.g., Kanigel 1997). Henry Ford's mass production technology was built on the use of such standardized tasks performed on standardized parts and materials (Hounshell 1984).

Rationalized work processes produce standardized products as well. Sometimes this is of value to end-users. When you're in a city far from home, for example, the presence of familiar franchised stores tells you a great deal about the quality of goods available. Sometimes, of course, standardization is a drawback because it reduces choice, but for intermediate or producers' goods standardized inputs means simpler production processes and lower costs in the next stage of production. Segregating and standardizing rationalization thus tend to reinforce one another. Segregation makes it easier to specify and enforce standards, while standardization makes it easier to segregate functions in different organizational units.

On the other hand, both segregation and standardization often trade off against coordinating rationalization, which seeks to make parts work together as efficiently as possible, minimizing waste and friction in the interests of a well-adapted system. This means fine-tuning interacting parts so as to make them accommodate one another's peculiarities. In doing so, coordinative tasks typically take advantage of local circumstances that provide an advantage. Such advantages, which economists call "positive externalities", are often incorporated into the production process. Warehouses, for example, tend to be located where transportation facilities come together.

Segregating and standardizing rationalization, by contrast, seek to improve over-all efficiency by removing dependencies or interactions among parts, and grouping each separate kind so that they can be processed uniformly at relatively low unit cost. They try to ignore or overcome local circumstances, and make work processes less dependent on positive externalities. As a result, coordinating rationalization processes can often be found in direct conflict with standardizing and segregating processes. For example, in writing computer

programs, it's often possible to improve the performance of a program by taking advantage of idiosyncracies in the design of the hardware or operating system on which the program is running. This form of coordinative rationalization is typically deprecated however, because it means that the program will run less well (or not at all) on other hardware or operating systems.

THE ROLE OF PROTOCOLS IN RATIONALIZATION

Segregation and standardization eliminate many kinds of local coordinative arrangement, but they do not eliminate the need to coordinate. Rather, they put it on a different basis. For example, if we employ a system of interchangeable parts to replace an artisan who fits variable parts together by trimming them, we need a system to ensure that the parts used are similar enough to make interchangeability practical. This typically means that new ways of coordinating work elsewhere in the production process-- new kinds of metawork-- must be implemented to compensate for the eliminated artisan. As I noted above, these new procedures tend to be formal, explicit, and standardized. They are, in fact, brackets aimed at replacing one way of coordinating work with another. More precisely, casual articulation work rooted in local and personal arrangements ("Honey, can you pick up some milk on the way home tonight?") is replaced with explicit coordination via formal mechanisms (the refrigerator adds milk to the shopping list, which is automatically transmitted to the supermarket via the Internet).

That is, increasing standardization eliminates some kinds of coordination work, but also provides the basis for new kinds of coordinative rationalization. Segregation and standardization may work against coordination *within* the scope of an particular activity, but they also enable new forms of coordination *among* activities as well. It is important to note two things about this process. First, local articulation is typically replaced by metawork. One example, becoming common now, is automatic updating of software on personal computer systems. Rather than ask users to keep track of which updates are needed, and when, and (perhaps) doing it themselves, perhaps correctly, an automatic process carries out this work. Second, the new kinds of coordinative rationalization are typically associated with increased reach over larger organizational, geographic, and temporal scales than was the case previously. In other words, if people can be convinced to do things the same way, then

knowledge of this similarity enables more effective (and/or simpler) coordination mechanisms. Again, automatic software updating is a good example.

The new kinds of coordinative rationalization are created through the development of standardized arrangements-- protocols-- that support metawork across many situations. Protocols abstract from many similar situations and specify a class of brackets or the mechanism for generating a bracket. The abstract character of protocols is critical, because abstraction means they can be represented, compared, combined, and manipulated as programs and data. This characteristic, in turn, enables coordinating many different sorts of tasks by similar means. Nothing illustrates this principle better than the Web itself, which answers almost any question by using standardized search services to supply appropriate URLs.

Protocols have become increasingly common and important as the Internet has developed; in a real sense, the success of the Internet is based on protocols such as TCP/IP, XML, SOAP, and RSS. In fact, in the last ten years specifying and exploiting new protocols has probably become the most important way of creating large-scale innovation in the industries concerned with information systems.

Protocols, with their standardized way of representing and interpreting information, provide a basis for replacing specialized local knowledge with general procedures. Using protocols means making things work in comparable ways in many situations. Tasks conducted in accordance with a protocol function adequately with a variety of other such tasks. They do this without imposing the need to worry about how others have their work arranged, and without respect to organizational boundaries.

Protocols thus enact a sharp distinction between form and content. Much of the metawork of inter-task coordination can thus be specified and organized without considering much of the detail of specific local circumstance. Typically, this is accomplished with the use of markup languages, application programming interfaces, and similar arrangements. Abstract protocols, with their focus on form rather than content, provide the basis for new kinds of bracket to organize the metawork associated with increased reach. Such protocols are particularly valuable because increased reach means that there more situations to be considered, and hence more variety and uncertainty in the local circumstances. The use of abstract protocols is not just a matter of inventing new standardized parts, like adding a new structural element to an erector set. Rather, it is like inventing a new class of objects which connect things that could not be connected before,

such as a new device that allows reliable connections between erector set parts and Lego blocks.

Consider, for example, some uses of airline flight information made practical by the growth of the Internet. Airlines and traffic controllers have always tracked flights in the air, and estimated arrival times have been available to the public via telephone for many years. Access to flight information via the Internet means that it's now possible for anyone to follow the pattern of arrivals and departures at local airports automatically. With this information, firms in the travel industry can coordinate their own work with airline flight times. Hotels can be aware that expected guests are arriving late. Limousine, shuttle, and taxi firms can get similar information for particular arrivals. They can also estimate changes in the flow of traffic to and from the airport, and thus improve their dispatching. This capacity is further improved if they can make use of location information (via GPS) from their own vehicles as well.

The abstraction of protocols thus enables many different sorts of systems to coordinate in ways which were previously impractical. For example, nobody ever actually wrote software to tie information about any hotel's room inventory to every airline's flight schedules. Yet every hotel now can do this. This same abstraction and generality also mean that many previously difficult local problems can now be solved easily. At the airport, for example, it is now much easier (in principle) for drivers and passengers to meet reliably despite bad weather, flight delays, traffic jams, and not knowing one another. Drivers and arriving passengers can now exchange photos and detailed directions for meeting via cell phone as passengers disembark.

These abstract protocols can be combined fairly easily to create applications (i.e., brackets) which did not exist before. This increase in standardized mix-and-match capacity means that people can come up with new arrangements that work across organizational boundaries, without the prior approval or even knowledge of their managements. Moreover, their innovations aren't confined to a single organization; often, new arrangements readily become available to everyone. This can have unexpected consequences on a large scale, as when the increased reach created this way forces changes on an entire industry. The clearest example of this to date is the emergence of peer-to-peer ("P2P") file sharing and its effect on the music recording industry.

For the most part, the new brackets enabled by protocols depend on the same computing innovations which are giving rise to increasing reach. Instead of relying on the

close integration of local circumstances, as articulation arrangements have traditionally done, they tend to be abstract and standardized. They also tend to rely on metawork rather than local articulation work. But the local articulation work must still be done.

LOCAL ARTICULATION AND THE LIMITS OF RATIONALIZATION

It's often difficult to rationalize (or even change) some local arrangements. Local problems are typically solved opportunistically. Casual improvisation over time becomes entrenched as local conventions, which are typically organized around the skills and tastes of particular people. As a result, local articulation arrangements are typically fine-tuned to the local situation, and dependent upon particular individuals and circumstances. Over time, new arrangements are added, each making use of idiosyncratic local conventions and further entrenching them. For this reason, local arrangements can be extremely effective, but it can be very difficult to understand or modify them.[2]

Local circumstances always have complexities that cannot be captured in a formal system, no matter how elaborate or forgiving it is. Hence, even as brackets are built to support local articulation work, there is always some articulation work beyond their reach. Well-designed information systems can help with these problems, but they can't eliminate them. That is: you can't build a repair kit that can anticipate all possible failures, and even if you could, you can't guarantee that it won't break just when it's needed. An important challenge for CSCW then, is understanding the nature of these limits.

The difficulties of doing so are greatly exacerbated by increasing reach, which forces a significant changes in the notion of "local". Traditionally, "local" referred to a particular organizational and/or geographic setting. With increasing reach however, it is now necessary to consider the joint performance of single activities or work-flows as "local" even though cooperating participants might be geographically and/or organizationally distributed. That is, "local articulation" (including "task-local" as well as "place-local" and "office-local") should refer to all the activities which require attention if a given task is to be carried out properly. Increasing reach means that colleagues may be interacting in different times of their daily cycles, or in different climates, or under different organizational policies or even legal systems. Moreover, they will be supported by different administrative services: computing, food, security, medical, janitorial… Indeed, these local contexts might be changing as

participants move between office and home, or from office to limousine to airport to plane and back.

With "local" more complicated than it used to be, analyzing and supporting articulation work has acquired new contingencies. Can brackets to support local articulation be built? What limits are there on doing so? In order to deal with these issues, it's necessary to refine the idea of local articulation work. Analytically, local articulation is really several different kinds of problem. I'll discuss two of them here: *customization*, which accommodates the particular and local to the general and global; and *reconciliation,* which accommodates the demands of different local stakeholders.

Customization: Accommodating the particular and local to the general and global

Abstract protocols present slots which must be filled in with appropriate local information. Providing this information is not necessarily a simple task; it requires interpretation on both sides in order to ensure suitable entries. For example, designers of data entry modules have long since learned to provide explicit alternatives wherever possible (e.g., "Sex: _____ (M/F)", rather than "Sex: _____"), templates to guide entry (e.g., "mm/dd/yyyy" for date fields) and help in order to ensure that data are entered in appropriate form. Of course, the guidance provided must itself be reasonable from the viewpoint of those providing the information. One kind of local articulation work thus consists of figuring out what the appropriate information is, gathering it, formatting it, and providing it. These tasks can prove to be formidable challenges in their own right. The difficulty people often face in filling out even relatively simple tax returns is a well-known example.

Of course, successful articulation consists of more than figuring out how to fill out forms effectively. This is important, but it is dependent on the coherence of the conceptual schemes used by both sides of the bracket. Where such schemes are known to all concerned, conventional, and commensurate with everyone's views and purposes, the problems of interpretation will be relatively minor. This does not mean that the consequences of a failure will be small. For example, accidentally interchanging metric and English units of measure is a small problem, in that it can be fixed quickly and cheaply if discovered in time. But the consequences of not noticing and fixing such an error can be very large. In 1999, for example, the Mars Climate Orbiter satellite was lost as a result of just such an error (NASA

1999). Where schemas are not understood, or not yet conventional, or incommensurate, the potential for some sort of failure increases. The chances of this happening increase with reach across task, organizational and geographical boundaries. So one major challenge to CSCW research is finding means of recognizing and describing when conceptual schemes are out of line with one another.

Customization also includes the work of making equipment and materials fit with local procedures. Strauss' original thinking about articulation work was motivated in part by his observation of health professionals working with specialized equipment in hospitals to make it fit the needs of particular cases (Strauss, et al 1985). Similar kinds of problem have been addressed by researchers studying how computer users "tailor" their software to fit well with local needs (e.g., Trigg and Bødker 1994).

Customization problems have concerned systems developers for many years. Indeed, such problems are at the heart of traditional research on computer-human interaction. The work of analyzing customization problems is worsened and sometimes transformed by increasing reach because traditional systems development has been done either within (and for) single organizations, or in ways which ignored local circumstance. Increasing reach means that systems interact with organizations and people in more intimate ways, at the same time that there are more opportunities for more things to go wrong in a greater number of ways. Such failures can easily lead to disaffection and conflict, thus turning problems of customization into problems of reconciliation.

Reconciling local participants

Most local articulation work consists of dealing with normal, expectable contingencies of coordination: the copier runs out of toner, someone calls in sick, an instrument needs calibrating, a customer is complaining about a routine problem, somebody has left the dirty coffee pot on the burner again-- the list is endless. Typically, there are established routines-- brackets-- for handling these contingencies. These may be as simple as "while you were out" message slips or fax cover sheets, or they may be elaborate procedures involving many people. Sometimes, these routines fail because the requirements imposed by one procedure come into conflict with those expected by another. When this happens, a bracket can reveal or become enmeshed in conflicts that were tacit or latent before it was installed. Indeed, the major point of one study of articulation work was that arrangements

for coordinating are themselves often the battlefields upon which contending participants work out their claims (Gerson and Star 1986).

This is the general reconciliation problem: Where there are multiple contending parties with differing viewpoints, interests, and concerns, some means must be found to reconcile their differences if their interaction is to continue. This is accomplished by aggregating the preferences of participants to form some kind of *collective* decision or policy that is reasonably efficient, effective, and equitable. Of course, choosing the governing criteria of efficiency, efficacy and equity is itself a reconciliation problem.

Consider the hypothetical example of a session at a scholarly meeting held in mid-winter at a hotel. As people come into the room, the room queries their personal computers or PDAs for their temperature preferences. The room then sets the room temperature to reflect the joint preferences of the attendees. From a technical point of view, this is a trivial problem: one chooses an algorithm to aggregate the individual preferences, expresses it a convenient programming language, and passes the result to the building's climate control system. However, any algorithm for aggregating preferences inevitably favors some preferences (or rather, preference holders) rather than others, so the issue of fairness necessarily arises. More generally, local conflicts are often reflected in decisions about bracket design and use. This is more likely to occur increase with increasing reach, if only because there are more kinds of local interest to reconcile.

Increasing reach means that the flow of everyday work thus comes to look like work in occupations (for example, medicine, sales, law, research, organized crime, and prostitution) where reconciliation problems are not always anticipatable. This is especially true where the client can't be commanded, but must be persuaded to cooperate. This is a very important point, because it makes for a qualitative difference in the kinds of problem-solving approaches which might work well. Tasks whose conduct crosses jurisdictional boundaries (whether organizational or territorial) don't have a ready formal or official means of reconciling differences. A disagreement among people in the same organization can be resolved by their common management, but a comparable disagreement across organizational boundaries only becomes more complex, more expensive, and no more certain of resolution, as managements become involved. This is a critical difference between systems oriented toward single participants and those oriented toward multiple independent participants. The means for resolving conflicts safely and reliably are very different. In these circumstances, support for local articulation means providing tools and procedures for

working with complex negotiations. There can be no general guarantee that these must be successful, or that agreement must be reached. Of course, the need to reconcile contending concerns is a consideration in every circumstance. But increasing reach intensifies and extends the problem in important ways.

Understanding this class of problems is a matter for basic research in all of the social sciences. One important and provocative approach to these problems comes from institutional economics, which has a significant line of work focused on these concerns. This has grown from the realization that the cost of shopping is a significant part of the costs of acquiring needed resources (Coase 1937, Williamson 1975, 1996). From this insight has grown an approach which examines the trade-offs between producing something "in house" on the one hand, and purchasing it on the other. Where there are few qualified suppliers (e.g., as when local knowledge is especially important), for example, producing a needed resource in house becomes more attractive.

Most of the conditions favoring use of the market, rather than in-house production, can be expressed in terms of the required metawork and articulation work. When needed metawork is standardized and readily available for purchase, and when local articulation work is easily and reliably done, then the appropriate coordination mechanisms are ordinary arrangements of buying and selling: proposals and quotes for fixed priced contracts, delivery schedules, and so on. When articulation work becomes complex and unreliable, and/or when metawork is not standardized, then the necessary negotiations are much more complex, and are more reliably handled in the context of a common administrative regime.

Models of each of the two major organizational forms, markets and hierarchies, have their strengths and weaknesses. Neither group of models can deal very effectively with the problems raised by increasing reach. This difficulty has spurred a search for other, intermediate, forms of association (e.g., Williamson 1985, 1996; Ostrom 1990; Ostrom et al. 1994; Ellickson 1991; and Rose 1994). The core research problem is to understand how different conventions and organizational characteristics affect the trade-offs among efficient allocation of resources, effective performance, and fairness. In particular, we are concerned with the implications of different ways of coordinating for reconciliation in the presence of increasing reach.

Articulation aimed at reconciling differences has an important and troublesome property. It cannot be cast as formalized mechanisms because the work itself involves justifying, designing, choosing, and enforcing just those very articulation procedures. This

gives reconciliation work a kind of self-referential character. Sometimes, this appears in practice as debates over legitimacy, flagged by such phrases as "…it's for your own protection", and "…who are you to give orders?". At other times, participants create workarounds which anticipate and adjust for inadequacies in local circumstances. For example, Gasser (1986) describes an engineering group which worked around defective software by running the program with false assumptions in a way that compensated for the design error. Sometimes, that is, it's necessary to go against policy, or standards, or even facts in order to get things done properly. This means that the brackets involved must be either flexible or subvertable. Above all, increasing reach forces us to consider reconciliation in situations where there is no common authority which can enforce a solution.

I'll mention three ways of organizing reconciliation here: cross-cutting ties among participants, participant review, and patronage. These are not the only possible arrangements; many forms are possible. These three organizational arrangements are neither market nor bureaucracy, and oppose the consequences of segregating and standardizing rationalization in many ways. They appear very frequently, and their possibilities and limits must be considered when thinking about brackets and information systems which embody them.

Cross-cutting ties

Reconciliation becomes more difficult when the problem lies, not in the conduct of tasks directly, but in conflict among participants. Social scientists have often found that conflicts are constrained and softened by "cross-cutting ties" (e.g., Gluckman 1965). That is, relations among friends, neighbors, relatives, and co-workers sometimes overlap considerably. When people in such situations come into conflict, there's a strong tendency for the quarrel to be softened by their mutual connections. In short, I'm less likely to do something rash to my neighbor's obnoxious pet, if my neighbor is also my cousin and my business partner. Similarly, I can more easily trust someone far away with my property and my interests if we have kin and tradition in common. Landa (1994) for example, has provided an extensive study and comparative analysis of long-distance trading networks based on ethnic and kinship ties. More generally, each person participates in multiple networks simultaneously: kin, co-workers, friends, professional colleagues, neighbors, and so

on. These networks often overlap in varying degrees. A problem in one network can sometimes be resolved by making use of connections through another overlapping network.[3]

In modern industry, organizational disputes are often mediated in this way. The process of side-stepping or working around official procedures often makes use of cross-cutting ties: common aspects of identity, common history, common memberships in voluntary associations, friendship bonds, patron-client ties, and common professional commitments are all used to undercut or overcome formal "bureaucratic" restrictions. The work is done by taking someone out for coffee or a drink, by "letting down one's hair" in private conversation, by offering personal advice, perhaps indirectly. To put something "on the record" would be to destroy the possibility of using the cross-cutting tie to solves the problem. In such situations, if management is called upon to notice there's a problem, it's too late to solve it in that form-- it's become another kind of problem. Sometimes then, cross-cutting ties conflict with explicit brackets. The effectiveness of using cross-cutting ties to deal with a reconciliation problem comes from personal connections, cumulative experience with particular individuals, and common bonds. Systems which do not take this into account often appear rigid, hence ineffective.

On the other hand, cross-cutting ties are not an unmixed blessing; they can be, and often are, are used to create or perpetuate differential access to benefits (Tilly 1998). For example, a group of friends may share knowledge of a valuable obscure resource (a good fishing site, for example) only among themselves. Or a group may capture the jobs in a particular industry or organization. Such ties are often used to obtain benefits outside the formal rules of bureaucratic or market organization, and are thus reviled as corruption or cronyism by those excluded. But this is simply to say, that there are no forms of social organization which are simultaneously efficient, effective, and fair by everyone's standards.

Participant review

Participant review is another form of organization which has been receiving increasing attention. By participant review, I mean the aggregation of individual votes or opinions to reach a collective decision. Peer review of academic papers and proposals is a familiar example; political elections are another. Simple popularity contests, in which the declaration of a winner has no effect on collective action, are not matters of participant review in this sense. The basic procedure is to have people rate performances or candidates

on some set of criteria. Ratings can then be summed or averaged to provide evaluative scores, which are then used to guide further action. Such rating systems are very flexible, because ratings can be weighted and aggregated in many ways. Moreover, collective policies can often be mapped onto the distribution of benefits in many ways as well. For example, certain categories of person (e.g., senior citizens, children, criminals, veterans) may be given preferential access to some goods and services.

Participant review in some form is useful for evaluating performances, policies, and performers in situations where there is no centralized authority which can make decisions and enforce them. The process seeks to capture some of the benefits of market mechanisms without suffering their drawbacks. It differs from markets because it does not use money (or some other store of value) as a medium of exchange. Merit accumulated in one setting therefore cannot, in general, be transferred to other settings in any direct way. For example, the fact that I am widely recognized as the second-best football player in my household gains me nothing vis-à-vis my colleagues in the International Society for Intellectual History.

One sophisticated system of participant review is the moderation system used by the Slashdot Web site (http://slashdot.org; see also Benkler 2002) to evaluate postings and comments on them. Another is Bugzilla, the defect tracking system developed for the Mozilla open-source browser project. This system is described by Sandusky, et al. (in press).[4]

The usefulness of participant review as a device for aggregating preferences and distributing evaluations varies with circumstances. It is more likely to work in situations where the relevant evaluation criteria are clearly understood and reproducible. Similarly, where raters have a stake in the quality of the outcome, the work of rating will be taken more seriously.

As with any mechanism, participant review has its limitations and defects. Such mechanisms can often be gamed or subverted in various ways; doing this to Google page ranks has become a minor sport. Such systems can also lose or distort the significance of local commitments and constraints. One way of dealing with weaknesses in the participant review system is to make it recursive. That is, the system can be used to review the performance of the participant review task itself (and by implication, the reviewers). Community pressure is thus brought to bear, in principle, on raters who are not careful and reliable. Another device is to limit the amount of rating which can be done, and then reward raters who achieve good reputations with additional opportunities. Some of the most successful participant review systems, such as the one used by Slashdot, make use of these

devices. Despite its drawbacks, participant review provides a flexible class of mechanisms for dealing with highly distributed evaluation problems. Moreover, these mechanisms lend themselves reasonably well to support with information systems.

Patronage

A patronage system is comprised of personal relations between a few relatively powerful patrons and their relatively weak clients.[5] Clients form a retinue or train that enhances the social position of the patron, and provides a pool of resources for the participants to draw on. The patron provides access to wealth, glory, or other scarce goods, and perhaps some kind of political protection as well. Such a system of relations aggregates preferences, organizes reputations, and regulates resource flow. Patronage systems can thus serve as organizations for reconciling differences among participants. Alternatively, a patronage system may perpetuate differences that span rivalrous clienteles ("factions"). These disputes are often muted by cross-cutting ties, as in "Romeo and Juliet" or "West Side Story".

The relations in a patronage system are personal. They exist between a particular person as patron, and particular others as clients who may, in turn, have clients of their own. A patronage system is thus made up of chains of influence, obligation, and loyalty, rather than choices among fungible commodities. Their structure is thus a matter of unique histories, not a system of formal relations. The commitments that clients and patrons make to one another do not, in general, transfer to others.

We see patronage in operation nowadays wherever we find uniquely accomplished people who can command substantial resources in their own right. The most familiar examples are the star systems characteristic of many branches of the entertainment industries. Stars can build substantial entourages of supporters and assistants. They can also influence the hiring and investment decisions of studios, teams, and other organizations in their industries. And of course, they also benefit from the deference paid to celebrity. Eric Raymond has noted that patronage plays a role in the support of highly distributed systems such as open source software development:

> "We may observe, in fact, that open-source firms hire star hackers for much the same reasons that universities hire star academics. In both cases, the

> practice is similar in mechanism and effect to the system of aristocratic patronage that funded most fine art until after the Industrial Revolution—a similarity of which some participants are fully aware." (Raymond 2001: 158)

The learned professions are a group of social worlds in which patronage relations flourish. Senior physicians, attorneys, professors, scientists, programmers and engineers control considerable resources through their positions on review committees of all sorts, and via the weight of their recommendations, formal and informal. The influence of an individual professional is acquired over time by converting personal reputation into referrals, appointments, assignment to plum projects, and other benefits. The personal reputation is acquired via participant review, formal and informal. Prestigious titles and awards serve as surrogate measures of reputation, but it is personal reputation which counts.

It isn't clear what effect the development of the Internet will have on patronage systems and their capacity to support and rationalize articulation work. One class of systems designed to support patronage has become quite popular: software used to organize social networks on the Web. There are now dozens of these services, and some of them apparently enroll many thousands of users.[6] Another potential source of Web-based patronage is the system of links among blogs. Since popular blogs can increase traffic at other blogs by linking to them, a link from such a blog can be an important resource. Whether these nascent forms of patronage will come to be significant outside their immediate context is not yet clear.

WHAT IS TO BE DONE?

Increasing reach means that many kinds of work are being changed in important ways. Traditional assumptions about the connections among tasks are changing, and new ways of working are being developed. As a result, the traditional assumptions that researchers have made about the relationships between technology and social organization now require careful re-thinking. Most research in CSCW has been conducted on the assumption of relatively low reach. For the most part, case studies have looked at single organizations, or single units within organizations. CSCW research must consider how computing supports (or doesn't support, or might support) tasks in which the co-workers have never met, which have no particular location, and whose tools have substantial

processing and communications capacities. This concluding section draws attention to some important research problems and suggests some strategies for development.

The changing division of labor in computing work

One important aspect of increased reach is its implications for the organization of systems development work itself. For many years, it has been conventional wisdom that systems development requires two broad classes of expertise. On one hand, it requires the skills of programming, system design and other aspects of information systems work. On the other hand, it requires "domain knowledge", i.e., expertise in the substantive work to be served by the system under development. Traditionally, the notion of domain knowledge has included needed information about the organizations in which an application would be used, as well as the technical aspects of the work supported.

The development of the Internet has extended the kinds of information systems knowledge needed, but it has also led to the addition of a third kind of requisite expertise: local knowledge. Describing local circumstances in suitable ways is itself a difficult task. The describer must have access to the information, and understand local circumstances well enough to represent them fairly and accurately without compromising needed confidentiality. At the same time, the describer must also be able to deal effectively with the systems development process. When the describers are system developers who have access to the ways local information is described, the problem is a manageable one. For systems of great reach, the system developers and the local describers are typically different people in different organizations. This can present insurmountable problems if relevant local circumstances cannot be described easily.

Designing and implementing brackets to support local articulation thus requires a great deal of specialized local knowledge. In situations where the tasks may be based in different organizations with different cultures, located in different parts of the world, and subject to different regulatory regimes, specifying a satisfactory bracket will be a formidable task, one that will require different ways of organizing project development teams.

Strategies for CSCW

It's clear that CSCW will be spending increased efforts on studying increased reach and its consequences. What help, at this stage, can CSCW research offer to systems developers

concerned with concrete projects? I start with three assumptions. First, no information system will ever completely eliminate the needed local articulation work for any setting. Second, it will be possible to design and implement many kinds of system that will aid local articulation work in many kinds of setting. Third, brackets should fail gracefully. That is, a system should be able to "let go" and leave the coordination problems to local people when circumstances make it impossible for the mechanism to function adequately.

A number of pertinent research programs and computing innovations are currently underway-- too many to permit a meaningful review here. These projects will provide a wealth of useful information to support the next round of research and experimental system development. Instead, I propose two broad rules of thumb: make customization easier and make reconciliation easier.

Make customization easier

The first suggestion is to make the work of customization easier. There are many ways to do this, and some of them are already part of standard practice. Few modern desk-top applications, for example (accounting applications aside), don't let users adjust the visual properties of the user interface such as screen colors or type face and font size. Information systems should go far beyond this, and support customizing local articulation tasks more generally.

Of course, no system can anticipate all the subtleties of a local context and its articulation problems. Rather than try to develop systems that will inevitably fail at this task, we should aim to turn local articulation experts into system developers. This can be done by building systems which support local bracket development much as integrated development environments support the development of programs. This idea rests on the distinction between domain expertise and local expertise. We can imagine local experts working with information systems experts to develop highly tailored systems using appropriate protocols.

One such system, for example, might support development of checklists to track completion of all the parts of complex projects. I am not thinking here of project management systems with their PERT charts, but rather a way to remind people and confirm that everyone has received needed resources, reminders, notices, and so on. As people in a particular situation discover that some participants need, say, more frequent reminders of approaching deadlines, this could easily be added as a heuristic. As experience

accumulates in each situation, it becomes clearer to participants which things need to be checked more rigorously, and which can be safely ignored. The essential point is that the design of such a system must be modifiable by users, so that changing circumstances and idiosyncrasies can be reflected in the system easily and quickly.

More generally, studies should focus on ways in which the use of standardized procedures meet local articulation work. This will directly support development of information systems that reduce the total amount of articulation required, and simplify the remainder.

Make reconciliation easier

Understanding reconciliation and the limits to support for it is certainly the most difficult problem facing CSCW research. There are several promising trends in the area. First, under the rubric of "social software", several different approaches to connecting people and facilitating negotiation are appearing and gaining considerable use. These are natural extensions of collective messaging or "bulletin board" systems which grew up in the 1980s. Major examples include group blogs, forums (in which participants post messages to public threads), and wikis (in which participants write and re-write joint texts). Many of the features characteristic of each form are easily added to other forms, so we can expect a period of enthusiastic experimentation in which many ways of organizing, supporting, and managing discussion and negotiation are tried.

For obvious reasons, the development of new protocols (as opposed the use of established ones) provides a rich place to study reconciliation. An especially interesting example is the system of facilities supporting the Atom standard. Atom is a protocol designed to replace and extend the functionality of the RSS (and kindred) protocols for disseminating updates to Web sites. Development began in late 2003, and was carried out entirely via negotiation among interested parties on the Internet. In the course of this negotiation, a mix of supporting Web sites, blogs, and wikis were established to support the development work.[7] The Bugzilla system for reporting bugs in the Mozilla project has similarly evolved into a substantial mechanisms for dealing with problems of reconciliation (e.g., Sandusky et al. In press). As this sort of arrangement becomes more frequent, we will learn a great deal about supporting reconciliation effectively.

Systems to support varieties of participant review also hold a great deal of promise, both as practical applications and as vehicles for the study of research problems. What kinds of evaluation criteria work best, and with what kinds of rating system? Should the distribution of rating points be restricted, and if so, how? What sorts of access control are useful or counter-productive, and under what circumstances? When is meta-review effective, and when does it take more trouble than it's worth? These are the sorts of question which will guide additional research and development.

Conclusion

Increasing reach based on the growth of the Internet has changed the way people work, the way we study work, and the role of computing in supporting work. This paper has focused on one important part of these changes: the nature of articulation work and the limits of computing support for it. Schmidt and Simone proposed the notion of coordination mechanism as a way to think about rationalizing articulation work in 1996. The impact of the Internet's growth since then requires that we think of articulation work (and hence, coordination mechanisms or brackets) in a more elaborate way.

It's important to distinguish between metawork (the work of organizing work) and local articulation work, the work of bringing together locally whatever is needed to carry out a task. Much metawork can be supported by computing technology in a fairly straightforward fashion. Moreover, the widespread adoption of many abstract standards and protocols makes it relatively easy to do so, at least by comparison with the pre-Internet situation.

Computing support in the form of brackets for local articulation presents a different and more complex, set of problems. The central point about systems to support local articulation is that, wherever the flow of interaction goes beyond the limits of a protocol, effective computing support becomes very difficult to construct. This will typically be so, since local articulation is ordinarily concerned with negotiating limits and boundaries, rights and obligations, evaluation standards and entitlements, morals and emotional commitments. Attempting to organize such interaction with a formal system is simply an invitation to game or subvert the system. Systems to support local articulation should thus be devoted to supporting the negotiation and related efforts which make up such work rather than the

work itself. Such systems are not likely to be brackets in the sense used here; they will guide the work only in the loosest sense.

[1] The history of computing is a substantial area of scholarship in its own right. Two useful overviews are Campbell-Kelly (2003) and Ceruzzi (2003). The development of information systems in the years before the emergence of computers is covered in Yates (1989).

[2] Becker (1982) provides a sociological analysis of entrenched practices. The philosophical foundations are discussed by Wimsatt (1986, 2001).

[3] The analysis of networks is a thriving part of sociology. Good introductions can be found in Watts 1999, 2003 and Burt 1992; Burt 2004 is particularly relevant to the problems discussed here.

[4] Slashdot users of the site register, so they are known to the system. They then submit short articles of interest, and comment on those selected for posting. Posts and comments are evaluated by users. Those who provide good service to the system gain "karma", and additional moderation privileges. The system has many complexities, (explained at http://slashdot.org/faq/com-mod.shtml#cm600) including "metamoderation" (explained at http://slashdot.org/faq/metamod.shtml). As with all mechanisms of governance, it is often contested. It does appear to work. Bugzilla is described at (http://www.bugzilla.org/about.html)

[5] Patronage has been analyzed in many contexts. Eisenstadt and Roniger (1984) provide a sociological overview. Schmidt et al. (1977) and Gellner and Waterbury (1977) are collections of studies from anthropology and political science. Clark (2000) argues convincingly that patronage was the dominant form of stratification in the eighteenth century. Biagioli (1993) provides a detailed analysis of the way patronage worked in early modern Italian science; Haskell (1980) discusses artistic patronage in the same period.

[6] A partial list of current services is maintained at http://socialsoftware.weblogsinc.com/entry/9817137581524458/

[7] The specification is housed as a wiki at: http://www.intertwingly.net/wiki/pie/FrontPage; important blogs which track the project are at http://www.atomenabled.org/ and http://danja.typepad.com/fecho/. A mailing list for discussion can be found at: http://www.imc.org/atom-syntax/index.html.

REFERENCES

Agre, P. E., and M. Rotenberg (Eds.). 1997. *Technology and Privacy: The New Landscape*. Cambridge, MA: MIT Press.

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

Benkler, Y. 2002. "Coase's Penguin, or, Linux and the Nature of the firm". *Yale Law Journal* 112: 1 –72. http://www.benkler.org/CoasesPenguin.html

Biagioli, M. 1993. *Galileo, Courtier: The Practice of Science in the Culture of Absolutism*. Chicago: University of Chicago Press.

Burt, R. S. 1992. *Structural Holes: The Social Structure of Competition*. Cambridge, MA: Harvard University Press.

Burt, R. S. 2004. "Structural holes and good ideas". *American Journal of Sociology* 110: 349-399.

Campbell-Kelly, M. 2003. *From Airline Reservations to Sonic the Hedgehog : A History of the Software Industry*. Cambridge, MA: MIT Press.

Ceruzzi, P. E. 2003. *A History of Modern Computing*. 2 ed. Cambridge, MA: MIT Press.

Clark, J. C. D. 2000. *English Society, 1660-1832: Religion, Ideology, and Politics During the Ancien Regime*. 2 ed. New York: Cambridge University Press.

Coase, R. H. 1937. "The nature of the firm". *Economica* N.S. 4: 386-405. Reprinted 1988 in *The Firm, the Market, and the Law*. Chicago: University of Chicago Press.

Eisenstadt, S. N., and L. Roniger. 1984. *Patrons, Clients, and Friends: Interpersonal Relations and the Structure of Trust in Society*. New York: Cambridge University Press.

Ellickson, R. C. 1991. *Order Without Law: How Neighbors Settle Disputes*. Cambridge, MA: Harvard University Press.

Gasser, L. 1986. "The integration of computing and routine work". *ACM Transactions on Office Information Systems* 4: 205-225.

Gellner, E., and J. Waterbury (Eds.). 1977. *Patrons and Clients in Mediterranean Societies*. London: Duckworth.

Gerson, E. M. 1998. *The American System of Research: Evolutionary Biology, 1890 - 1950*. Unpublished PhD dissertation. Chicago: University of Chicago Department of Sociology.

Gerson, E. M., and S. L. Star. 1986. "Analyzing due process in the workplace". *ACM Transactions on Office Information Systems* 4: 257-270.

Gluckman, Max. 1965. *Politics, Law, and Ritual in Tribal Society*. Chicago: Aldine.

Haskell, F. 1980. *Patrons and Painters: A Study in the Relations Between Italian Art and Society in the Age of the Baroque*. 2 ed. New Haven: Yale University Press.

Hewitt, C. E. 1977. "Viewing control structures as patterns of passing messages". *Artificial Intelligence* 8: 323-364.

Hewitt, C. E. 1991. "Open information system semantics for distributed artificial intelligence". *Artificial Intelligence* 47: 79-106.

Hounshell, D. A. 1984. *From the American System to Mass Production, 1800-1932: The Development of Manufacturing Technology in the United States*. Baltimore: Johns Hopkins University Press.

Kanigel, R. 1997. *The One Best Way: Frederick Winslow Taylor and the Enigma of Efficiency*. New York: Viking.

Landa, J. T. 1994. *Trust, Ethnicity, and Identity: Beyond the New Institutional Economics of Ethnic Trading Networks, Contract Law, and Gift Exchange*. Ann Arbor: University of Michigan Press.

Lawrence, P. R., and J. W. Lorsch. 1967. *Organization and Environment: Managing*

Differentiation and Integration. Boston: Harvard Business School Press.
Malone, T. and K. Crowston. 1994. "The interdisciplinary study of coordination". ACM *Computing Surveys* 26: 87 - 119
NASA. 1999. "Mars Climate Orbiter Mishap Investigation Board Phase I Report". Washington: National Aeronautics and Space Administration. ftp://ftp.hq.nasa.gov/pub/pao/reports/1999/MCO_report.pdf
Ostrom, E. 1990. *Governing the Commons: The Evolution of Institutions for Collective Action*. New York: Cambridge University Press.
Ostrom, E., R. Gardner, and J. Walker. 1994. *Rules, Games, and Common-Pool Resources*. Ann Arbor: University of Michigan Press.
Raymond, E. S. 2001. *The Cathedral and the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*. Cambridge, MA: O'Reilly.
Rose, C. 1994. *Property and Persuasion: Essays on the History, Theory, and Rhetoric of Ownership.* Boulder, CO: Westview Press.
Sandusky, R.J., L. Gasser, and G. Ripoche. In press. "Bug Report Networks: Varieties, Strategies, and Impacts in a F/OSS Development Community."
Schmidt, K. and C. Simone. 1996. "Coordination mechanisms: Toward a conceptual foundation of CSCW system design". *Computer Supported Cooperative Work* 5: 155–200.
Schmidt, S. W., J. C. Scott, C. Landé, and L. Guasti (Eds.). 1977. *Friends, Followers, and Factions: A Reader in Political Clientelism*. Berkeley: University of California Press.
Stinchcombe, A. 1990. *Information and Organizations*. Berkeley: University of California Press.
Strauss, A. L. 1987. "The articulation of project work: An organizational process". *Sociological Quarterly* 29: 163–178.
Strauss, A. L. 1991. *Creating Sociological Awareness*. New Brunswick, NJ: Transaction Books.
Strauss, A. L., S. Fagerhaugh, B. Suczek, and C. Weiner. 1985. *The Social Organization of Medical Work*. Chicago: University of Chicago Press.
Thompson, J. D. 1967. *Organizations in Action: Social Science Bases of Administrative Theory*. New York: McGraw-Hill.
Tilly, C. 1998. *Durable Inequality*. Berkeley: University of California Press.
Trigg, R. H., and S. Bødker. 1994. "From implementation to design: tailoring and the emergence of systematization in CSCW." *CSCW 94* 45-54.
Watts, D. J. 1999. *Small worlds : the dynamics of networks between order and randomness*. Princeton, N.J.: Princeton University Press.
Watts, D. J. 2003. *Six Degrees: The Science of a Connected Age*. New York: Norton.
Williamson, O. E. 1975. *Markets and Hierarchies*. New York: Free Press.
Williamson, O. E. 1985. *The Economic Institutions of Capitalism: Firms, Markets, Relational Contracting*. New York: Free Press.
Williamson, O. E. 1996. *Mechanisms of Governance*. New York: Oxford University Press.
Wimsatt, W. C. 1986. "Developmental constraints, generative entrenchment, and the innate-acquired distinction." Pp. 185–208 in P. W. Bechtel (Ed.). *Integrating Scientific Disciplines*. Dordrecht: Martinus Nijhoff.
Wimsatt, W. C. 2001. "Generative entrenchment and the developmental systems approach to evolutionary processes." Pp. 219-238 in S. Oyama, Paul E. Griffiths and R. D. Gray (Eds.). *Cycles of Contingency: Developmental Systems and Evolution.* Cambridge, MA: MIT.
Yates, J. 1989. *Control Through Communication: The Rise of System in American Management*. Baltimore: Johns Hopkins University Press.