NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage Nous avons tout fait pour assurer une qualité supérieure de reproduc tion

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser a désirer, surtout si les pages originales ont été dactylogra phiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure

Canadä

Conversation in the software development process

Margot Hovey

A Thesis

in

The Department

of

Communication Studies

Presented in Partial Fulfilment of the Requirements

for the Degree of Master of Arts at

Concordia University

Montréal, Québec, Canada

April 1992

© Margot Hovey, 1992

Canada

Abstract

Conversation in the software development process
Margot Hovey

The introduction of communication technology is changing organizations as we know them. Current research in Computer Supported Cooperative Work (CSCW) is addressing how people use technology in a cooperative work environment. Also, research in organizational communications suggests that organizations should be interpreted in a transactional framework where organizations are based on the processes of human transaction that occur in their formation. These areas of research are converging in the communicative patterns implicit in how people use technology in a cooperative work environment. Using an extended case-study research method, this thesis is devoted to a critical examination of the relationship between the automation of administrative processes and the communication patterns of an organization.

This study focuses on how the conceptualization of the administrative processes changed through the phases of the software development process. The analytical construct of the administrative processes is established through textual analyses of documents produced during the planning phase. These analyses reveal traditional computer science practices and assumptions. The impact of conversation on the conceptualization of administrative processes is revealed through the interactions between the system designers and the user representatives during the development phase.

Through the examination of the relationship between the automation of administrative processes and the communication patterns of an organization, it is revealed that the conceptualization of administrative processes was redefined to include the social dynamics of the organization: a dynamic organizational structure, variable inter-departmental

relations, evolving job responsibilities, the use of local information, divergent administrative practices, and multiple user groups. In conclusion, recommendations are presented for incorporating the social dynamics of an organization in the planning phase of software development process.

Acknowledgements

It is rare to meet a teacher whose encouragement moves you to explore your own ideas. It is rarer still, to meet a teacher whose guidance keeps you marching to the tune in your own head. This thesis is based on many hours of conversation with Professor J. Taylor, Université de Montréal. For his patience and support, I am truly grateful.

Professor W. Gilsdorf and Concordia University have sustained my quest for knowledge and have made this education possible. The Communications department has provided the framework for the completion of this thesis. All the members of the department have fuelled this process.

My mother and her mother (and all the mothers back to Nellie McClung) provided the inspiration to shed light on social realities.

Members of R&DTel. provided useful feedback in maintaining the anonymity of this work.

My partner, Richard Minichiello, is by my side at all times.

Contents

Introduction

The completion of this study occurs after two years of direct observation, close participation and deep contemplation. This study has been based on the experience of the development of an automated administration system for a large telecommunications company.

Intention of this chapter

The intention of this chapter is to provide an introduction to the major components of this thesis. These include: Background to the case study, Research problematic, History/methodology, Rationale, Background on the corporate family, Background on the family dynamics, Background on the players, and the Organization of this thesis.

Background to the case study

The software application from the outset seemed to be a manageable task. The objectives were well defined, the funding was available, the customer's management believed the project objectives were in line with their business plans, and the software experts had been assigned. In short, all the project management plans and resources were in place. Over the time that I was involved in this project, deadlines and objectives were constantly redefined. There was real confusion when speaking to other project players about the project. All the signs and indicators seemed to imply that the project was doomed. Nevertheless, eventually the administrative software was actually produced. This study seeks to make sense of what happened; why the rational project lost control yet survived as a project and today is in field trial where it is being tested in the customer's organization.

Research problematic

The evolution of information technologies in the past decade has given scholars in organizational communications much food for thought. Through an extended case study, this research project will examine how the conceptualization of administrative processes evolved through the interaction between software designers and user representatives to reflect the communication processes of an organization in the information age.

To provide an explanation of the construction of the concepts of the administrative processes in this software development project, there are two primary objects of inquiry related to this problematic. The first is to determine how the concept of the administrative processes was constructed during the systems design phase. The second is to show how this concept of the administrative processes was modified to incorporate the social dynamics of the organization through conversation during implementation.

Examination of the issues that are related to these objects of inquiry will provide an explanation of the concept of administration processes. These issues include: how traditional systems analysis constructs a formal vision of organizations; how the formal vision of organizations equates administration processes with information processes; how the assumptions made by system designers about the translation of a paper-based administration into an automated administration effected the conceptualization of administrative processes; how the social dynamics of the administration processes must be accommodated in software design for implementation to be successful; and finally, the effect of conversation on the software development process.

This study examines how views of an organization are constructed and maintained through the conceptualization of administrative processes in the software design process. Through this examination, the relationship between information technologies and organizations can

be explored in a new way. By investigating the conceptualization of administrative processes during both the design and implementation phases, this study will consider the interrelationships between information technologies and organizational communications.

History/methodology

A case-study approach has been used as the fundamental method of research. The duration of the case extended over two years. The choice of research methodologies in this case emerged through my participation in this project. These methodologies are primarily participant observation and textual analysis.

Participant observation

Most of the research is conducted as a participant in the development of a software project. I was an employee at the software development site and worked on different aspects of this project for two years. This project is based on my experience as a technical writer and as an academic observer.

Through participant observation I was involved in documenting many end-user documents on this project. In my capacity as a technical writer on this project, I was privy to many meetings and conversations with the software designers and user representatives responsible for this project. These conversations took place between December '90 and July '91. During the evenings and after significant conversations with software designers and user representatives, I took over two hundred pages of descriptive notes.

As well, I had access to many corporate documents (eg. user-needs analysis, administrative procedures and memos) in my participation on this project. Relevant organizational texts that were produced during this timeframe are paraphrased and included as research data.

Textual analyses

I have provided textual analyses of the documents that were produced during the software design process. Corporate documents are analyzed, but not included, in this thesis. All corporate information is considered proprietary and will, therefore, not be available for review.

Organization of texts

The format of the thesis is a historical narrative. I will tell the tale of the development of the software in a chronological order. *Chapter 2: Analytic foundation* includes analyses of texts that preceded my involvement in this project. *Chapter 3: Talking through development* includes "snatches" of conversations and notes taken during my participation. In chronicling the evolution of software, I will reveal how the context of systems design constructed a framework in which the conversational events of the software development can be understood.

Rationale

In this chronicle, I will show that the concept of administrative processes changed through the software development cycle. The rationale for this study is that it contributes to current research in Computer Supported Cooperative Work (CSCW), attempts to balance objective with subjective methodologies in organizational communications, and provides practical advice for administrators.

This study benefits CSCW research in three ways. First, this case study is based on research in an actual telecommunications company. Therefore, the group dynamics are genuine and are representative of those found in any large organization. In this sense, the organizational dynamics of this research is typical to organizations anywhere.

Second, this case study extends Kyng's (1991) concept of mutual learning to incorporate conversation. Kyng postulates mutual learning as a way for software designers and end-users to get to know each other (see *Chapter 1: Theoretical framework*). Mutual learning, for Kyng, is based on working together in software design. His approach is oriented towards cooperating in tasks. He does not, however, approach the effect of conversation between software designers and users (or representatives of users). In this case study, I will show how system designers and the user representatives got to know each other through conversation. Through conversation, the reformulation of the administrative processes was made possible.

Third, the implementation of administrative software in this case study, extends the concepts of cooperation in CSCW research. Cooperation has been defined by the act of working together on a specific task using some kind of technology. In this sense, technology has united people and therefore, has been the basis of cooperation. Yet, the administration of an organization depends on the cooperation of people working together. The day-to-day operations of an organization rely on the cooperation of people working together. In this study, it is the activities involved in administration, not technology, that are the basis of cooperation. The technology is developed and used for the purpose of administration. This case brings CSCW research into the administration of organizations. In this sense, it is applicable to all organizations. Moreover, this case relates the administration of an organization to current CSCW research.

Weick says, "A prototype case study could demonstrate the value of the interpretive approach by showing how a myth, image, or *conversational pattern* constrains and is constrained by size, technology, *formalization*, and culture (emphasis added)."[1] In this

---

[1] Weick K., "Research Agenda", in L. Putman and M. Pacanowsky (Ed.), Communication and Organization an Interpretive Approach, Sage Publications, 1983, pp.14.

5

case I will show that the conversational pattern between software designers and users is constrained by the formalization of systems analysis in traditional computer science. The conduct of this research, the methodologies used in this case study provide a unique balance of textual analysis of documents found in organizations along with participant observation. These research techniques were selected because they were the most fitting to the problems at hand. Yet, by analyzing the texts produced through systems analysis, I will trace how they provided a framework for the conversations in the software development process. Moreover, the analysis of these texts provides an objective framework for the subjective presentation of how the construct of administrative processes that was formalized in systems analysis changed as a result of conversation between software designers and user representatives. In this case, the analysis of texts provides a skeleton for the participant-observation technique.

On a practical note, this case study is relevant to other telecommunications organizations. Indeed, it is relevant to any other organizations in their implementation of administrative communications technologies. The conceptualization and construction of administrative processes is a necessary step for any organization that considers automating its administration. The findings of this research could be used by managers as a method of feedback for custom systems design.

Background on the corporate family and family dynamics[2]

Before this case is presented, I will provide some background information on the corporate family, the dynamics within the family, and the players on this project.

The players involved in this project are from two distinct corporate entities. The software planners and developers are employed by R&D Tel., a research and development organization. The customer, TelCorp., is the telecommunications company that has commissioned the development of the administrative software. A third corporate member of the family is dedicated to manufacturing telecommunications products. In the development of the administrative software, the manufacturing arm was not represented. This is not to say, though, that they would not be involved in the sale of an administrative software package to other telecommunications companies in the future. Products developed as a result of the relationship between R&D Tel. and TelCorp. are often used as a test case. If products are successfully developed by R&D Tel. for TelCorp. they are likely to be successfully marketed by the manufacturing company to similar telecommunications companies. The businesses of the three corporate entities are distinct, yet, mutually supportive. The corporate family, is vertically integrated in this regard.

The research and development organization has been a distinct corporate entity for twenty-five years. R&D Tel. evolved from a research and development department in the manufacturing company. It was created to allow for a structure, management, and corporate culture that was conducive to the "creative" demands of technology research and development. Members from both organizations relate to their own organization, not the family of organizations that make up the larger corporate family. For this reason, each organization is treated as a separate entity in this study.

---

[2] To maintain anonymity, the names of persons, products, and organizations have been created. Any resemblance to existing persons, products, or organizations is coincidental.

Family dynamics

There is no formal structure within the corporate family to guide projects of this sort. People are simply brought together based on their specific knowledge or expertise. They are brought together on a project basis. Each person is responsible for a certain role and represents the part of the corporate structure that they come from. Each person is accountable for his/her input to his/her own superiors. This is not to say that nobody is responsible for the project. But the responsibility includes measures of success related to the project. This responsibility does not include any of the human dynamics of people working together.

Players

The following section identifies how the software planning and development activities are positioned in their respective organizational structures. The following questions are addressed; who are the players involved in the planning, development and implementation of this project?; and how are these players positioned in their respective organizational structures?

The software planners

The players in R&D Tel. who were involved in software analysis or planning activities for this project are a part of a planning department. There are several planning departments in R&D Tel. Planning departments are generally co-located with development departments of this organization. The planning department that was responsible for the initiation of this project, however, is dedicated to planning new technologies in the operations of telecommunications companies.

R&D Tel. is divided into eight functional divisions. Each division is divided into approximately 26 subdivisions. Each subdivision is a cluster of activities with a common

theme. The software planners involved in this project are in a subdivision whose theme is network operations.

The mandate of this group of software planners is to define new technologies in the operations of telecommunications companies. Based on their knowledge of the operations of these companies and existing technologies, they investigate and define technologies to be used in the operations of telecommunications companies. The software planners define technologies up to the point where a group of software developers can take it over to create them.

New planning groups are evolving at a fast pace. There is great interaction between new and old planning groups. The reason for this is that they collaborate on the convergence and distinction of new technologies. Interaction also occurs because they are involved in the same kind of work.

The software planners are interested in creating business opportunities for the research and development organization. They are the technology-dreamers. Based on their mandate, they are interested in anything that will fuel the dreams of technology. For this planning group, in particular, their main interests are the current operations of the telecommunications company. The rationale for the current operations needs to be understood to conceptualize how technologies can be applied to accommodate the current operations. The pragmatics of the operations is their main interest. For with this information, coupled with a vision of other technologies that are being created in the research organization, they can dream on the right track.

9

The software planners were the authors of the initial project specification documents. They conducted research into the customer's operations. They wrote the documents that would later be blueprints in the design of the administrative software. These documents are analyzed in *Chapter 2: analytic foundation.*

The software designers

The software designers involved in the Administration software project are a part of the network operations subdivision at R&D Tel. The mandate of this group of software designers is to develop new technologies in the operations of telecommunications companies.

Once a project has been planned, the software designers take the project over from the software planners. Software designers rotate between development groups approximately every two years at R&D Tel. This is a part of their career planning.

Typically, the software designers have at least one university degree in computer science. The company is known for hiring new graduates directly from the university campuses. The average age on a project development team is usually around 30.

The software designers are well versed in the traditional methods of systems analysis and design. Since they usually have had few actual software development projects under their belts, they can be expected to rely on school-based training.

The user representatives

The user representatives were responsible for planning the project within their own organization. Unlike the software planners, their sole function is not planning. In this instance, the customer planners are a part of an operational methods group.

The structure of TelCorp. is based on functional lines. There are a total of three primary functional line groups. These three line groups are commonly referred to as he "families". Each family is hierarchically structured into tiers. Each tier represents a level in the hierarchy. There are five tiers in each family. There are methods groups for each family in TelCorp. The structure of TelCorp. is based on the technical operations of telecommunications. The methods department oversees the administrative methods of its' family.

The methods department is responsible for standardizing the operations of its own family. Operations are standardized by determining the most cost-effective ways of doing things for the organization as a whole. As a result, the methods departments conduct economic studies frequently throughout the year. They are also responsible for writing corporate practices and procedures that specify how specific tasks are completed in the organization.

The methods people have a lot of contact with members of the line groups. The studies that they perform involve talking to line members to collect information. In the case of this project, the line members are the administrative software "users". Once the software development was underway, the methods people relied on previous contacts, and initiated new contacts to collect information about the operations of the families.

The user representatives came from three methods groups at TelCorp. The Traffic methods department were "prime" for this project. They had responsibility for the coordination of resources to successfully implement the software in the organization. Their interests in this project were directly related to their job performance.

There are three main players in the Traffic Methods department. Jeff B. is the Tier 4 Manager. and Monique R. reports to him. The Transmission Methods department plays a secondary role in the project. Andy F. is the Tier 4 Manager and Nadia P. reports to him. The Network Maintenance also plays a secondary role in the project. Jonothon T. is the Tier 4 Manager and Geoff W. reports to him.

The users

The actual users of this software are the line members. The users are contacted throughout the project. They are, however, not involved in the systems analysis or design activities. Consequently, I did not have direct access to the users. They are information-providers only. Key contacts were established during the planning of the project. These same people were involved in all the major steps of the project. This project sets them apart from their peers.

Organization of this thesis

In *Chapter 1: theoretical framework* I will situate this case study within a theoretical construct of organizations that addresses the relationship between administrative processes and the communication patterns of an organization. This construct is composed of a view of organizations and current CSCW research that addresses how system designers and users interact.

In *Chapter 2: analytic foundation* I will examine what the analytic foundation of the concept of administrative processes was during the planning stage. The analytic foundation of the concept of administrative processes includes: a) the context that the texts were produced in; b) the methodologies used to interpret administrative information; c) the dominant paradigm in which these analyses were performed; d) the assumptions made in the analysis of administrative information, and e) the role of background knowledge in constructing the concept of administrative processes.

In *Chapter 3: talking about development* I will chronicle the software development cycle through the conversations that I participated in to show how administrative processes were redefined to include a dynamic organizational structure, variable inter-departmental relations, evolving job responsibilities, the use of local information, divergent administrative practices, and multiple user groups. At the close of the development cycle, administrative processes were viewed from a socio-technical view of the organization.

In *Chapter 4: conclusion* I will present my conclusions. I will summarize the findings of this study and apply them to the concept of mutual learning as defined by Kyng. Finally, I will present recommendations to the traditional stages of systems analysis and systems design which could be modified to incorporate the findings of this case study and complement the mutual learning approach to systems development.

Chapter 1: Theoretical framework

Introduction

Over the past fifteen years, interest in the social implications of technology has grown steadily (Olson and Lucas 1982, Lucas 1986, Marcus and Robey 1988, etc.). The common themes of research in the social implications of technology are in the relationship of technology to organizational structure (Leifer 1985, Markus and Robey 1988, George and King 1991) and in the relationship of technology and organizational change (Child, Gunter, and Kiesler 1987, Robey 1981). These areas of research are legitimate in their own right. However, they do not provide a framework from which the research problematic of this study can be addressed, namely how the conceptualization of administrative processes evolved through the interaction between software designers and user representatives to reflect the communication processes of an organization. There are two theoretical elements in this literature that are missing. First, the relationship between administrative processes and the communication patterns of an organization. Second, the interactions between system designers and users, or their representatives, are not addressed. In the absence of this theoretical framework, I believe it is impossible to understand how the concept of administrative processes changed through the interactions between systems designers and user representatives.

Intention of this chapter

The intention of this chapter is to present a theoretical construct of organizations that addresses the relationship between administrative processes and the communication patterns of an organization. This construct is composed of a view of organizations and current CSCW research that addresses how system designers and users interact.

Organization of this chapter

In "Communications in organizations," I will present a theoretical construct that illuminates the relationship between administrative processes and the communication patterns of an organization. In "Interactions between software designers and user representatives", I will give a general overview of CSCW research and highlight some of the issues that relate to this case.

Communications in organizations

The relationship between the administrative processes and the communication patterns of an organization can be addressed through the theoretical construct of an organization as presented by J. Taylor. Taylor's concept of organization "is based on the processes of human transaction that occur in their formation."

> "When I think of an organization, for example, I don't have in mind some kind of body, such as a machine, or an organism, or a brain, or an actor with a personality; I am conceiving of it instead as a universe of communication. An organization, as I visualize it, is nothing but a fabric of communication: a collection of people talking, writing, and transacting with each other. That, when you come down to what can actually be observed and described, is reality. Everything beyond those processes that we attribute to organization is a product of our imaging. From this perception a simple proposition follows: *changes in the technology of communication may conceivably change the pattern of how people communicate with each other*. And when this happens, you have in effect altered the structure - indeed the very nature of the organization." [3]

Taylor conceives of organizations as communication patterns. This speaks to what other theorists have missed in studying the implementation of communicative technologies. What is important about the implementation of new technologies is how communication patterns are changed.

---

[3] Taylor, J., The Vulnerable Fortress (Forthcoming), pp. 2.

The implementation of administration technologies is a special case. "Administer" is defined as *"to manage or direct the affairs of a government or institution"*[4]. Administration is closely intertwined with all aspects of the business of the organization. The way that administration is conducted is mirrored in the construction of the organization. The implementation of administrative technology can be expected to be felt in all aspects of the organization.

The automation of administration has the potential to change most of the communication patterns within the organization. Communication patterns exist between co-workers, between staff and management, between departments, and, with the implementation of communications technologies, between system designers and users, or their representatives.

Taylor's view of organizations as communication patterns provides the theoretical framework to examine the relationship between automated administration and the communication patterns of the organization. As such, it provides a framework to specifically address the interactions between software designers and user representatives.

Interactions between software designers and user representatives

Recent research in Computer Supported Cooperative Work (CSCW) approaches the relationship between technology and the communicative patterns of an organization through research conducted in cooperative work settings. In this study, this research is relevant for two reasons. First, CSCW research recognizes the existence of communicative patterns between people in a work setting. Second, a part of this research specifically addresses the interactions between software designers and user representatives in the software design process.

---

[4]Webster's New World Dictionary, pp. 18.

16

The following provides an overview of CSCW research to describe how CSCW research recognizes the existence of communicative patterns between people in a work setting.

Overview of CSCW research

Generally speaking, CSCW research examines the interrelationships between technology and a cooperative work setting. The existence of a cooperative work setting assumes that people are working with technology and communicating together. Currently, there are two major trends in CSCW research: research that focuses on the technology that structures cooperative work (Francik, Rudman, Cooper, Levine 1991) and research that focuses on the role of technology with respect to cooperation (Kling, Norman, Kyng 1991). The following briefly describes both trends of CSCW research and how they relate to this study.

Focus on technology

In technology-oriented CSCW research studies, technology actually defines the work group. For instance, video conference technology will bring together people in a meeting to discuss specific issues. I have witnessed this type of application in organizations that have departments in different locations. Video conferencing is used for weekly meetings. Another example of research in this area is technology that brings people together to work on a specific project who are in geographically scattered locations. Electronic mail has frequently been used in these studies.

Methodological shortcomings of technology-oriented CSCW research

I have chosen to take a case study approach to the implementation of the Administration software in an actual work setting because of the methodological shortcomings of technology-oriented CSCW research. Rob Kling notes, that "Researchers often illustrate their CSCW systems with hypothetical groups, or groups drawn from universities or industrial labs where reciprocal give and take is commonplace. Consequently, they tend

not to observe (or report) how status differences, organizational control systems, and hierarchy are reflected in the use of their groupware prototypes."[5] With a focus on cooperative *tools*, researchers are unable to fully test the use of the tools in a 'typical' group situation. This case will directly address the group dynamics that occur in an actual organization.

Focus on cooperation

CSCW research that is focussed on cooperation in its application asks the following question: how is technology used to enhance cooperation? In cooperation-oriented CSCW research, support is gaining for the participation of end-users in the software design process. The December issue of Communications of the ACM is devoted to current research in CSCW. In this issue, D. Norman summarizes this approach as follows:

> "In *traditional computer science, one worries about data structures and coding practices*, about ways to ensure that the program does what it promised to do, ways of proving that the code works, ways of synchronizing procedures within complex data and computational streams. All this is important and essential, but it has little to do with the issues that concern program users. Computer systems intended to aid people, especially groups of people, must be built to fit the needs of those people. And there is no way that a system can work well with people, especially collaborative groups, without a deep fundamental understanding of people and groups. *This is usually not the sort of skill taught in computer science departments* (emphasis added)."[6]

The teaching of computer science, then, bypasses skills allowing an understanding of people and groups. The end result is that, "Too often technology is constructed for its own merits, independent of its users, independent of the user...It is not appropriate when the

[5] Kling, R.,"Cooperation, Coordination and Control in Computer-Supported Work", Communications of the ACM, December 1991, pp. 85.
[6] Norman, D. "Collaborative Computing: Collaboration First, Computing Second", Communications of the ACM, December 1991, pp. 88-90.

whole point of technology is to act as a direct aid to people's work activities. When this is the case, then *the technology will only succeed if the people and the activities are very well understood.*(emphasis added)"[7]

An understanding of people and groups may not be directly addressed in traditional computer science. Yet, an image of the situation (organization) must be operative or assumed in the system design of collaborative systems. And through this view of the organization, some assumptions about the users and their activities of the system can be inferred. Even in the most extreme case, where a system is designed without any users in mind, who they are and what they do is inferred in the very design of the system. What Norman misses, is *if the conceptualization of the people and their activities are inaccurate, the implementation of the system will not succeed.*

Morten Kyng is another contributor in the December issue of Communications of the ACM. He talks about how computer applications can be designed with the user to integrate cooperation into software products:

> "To emphasize the cooperation between designers and users in early project activities, we adopt the term "mutual learning" to replace the widely used term "analysis" *Mutual learning implies that designers learn about the application area and users learn about new technical possibilities.* Furthermore, mutual learning encompasses the development and learning of new ways of cooperating that may be required of users and designers (emphasis added)."[8]

Kyng suggests different techniques for bringing software designers together with end-users in a collaborative effort. For instance, "Another example of system designer bias in

---

[7] Norman, D., "Collaborative Computing: Collaboration First, Computing Second", Communications of the ACM, December 1991, pp. 89.
[8]Kyng, M., "Designing for Cooperation", Communications of the ACM, December 1991, pp.70.

design work is the use of the concept information system. During our first cooperative design projects we realized that most users do not think about the applications they use as information systems."[9] This approach, Kyng believes, has "the potential of taking an important step from implementation tools towards user accessible tools."

To engage in mutual learning, each party must understand the other's point-of-view. Kyng concedes that a major challenge "relates to the ability to move between the worlds of the users and the designers". Kyng's study relates many useful techniques for bringing designers and users together in a project situation. *He does not, though, indicate what is involved in moving between these two worlds.*

How this case fits into CSCW research

What is missing from this research in CSCW, is how designers construct and maintain their world or how users construct and maintain theirs in the systems design process. Through a pragmatic approach to systems design in the case I am about to present, it will be possible to examine the role of conversation between software designers and user representatives in bridging these two worlds. This type of understanding, I believe, will illuminate other ways designers and users can exchange their views of, if not the world, then of the social situation, or organization, that the software is designed for. Both software designers and user representatives have visions of the use of technology in their situation. For technology to be successfully implemented in a cooperative work-setting, the software designers and the user representatives must create a shared vision of the organization that the technology is designed for.

In *Chapter 2: analytic foundation,* I will present how the system designers' vision of the organization was constructed in the concept of administrative processes in this case. In

---

[9] Kyng, M., "Designing for Cooperation", Communications of the ACM, December 1991, pp.70.

Chapter 3: talking through development, I will describe how the system designers and user representatives created a shared vision of the organization through the conceptualization of administrative processes through conversation.

Chapter 2: Analytic foundation

Introduction

The intention of this thesis is to show that the concept of the administrative processes changed from the planning context to the development context. To show how this change occurred, I will first describe how the administrative processes were conceptualized in the planning phase. This chapter will describe how the administrative processes were conceptualized during the planning of the Administration software. This will provide a framework for *Chapter 3: Talking through development*, where I will show how the concept of the administrative processes changed through conversation.

Analysis of texts

The approach that I will take to show how the concept of administrative processes was constructed is through the analysis of the texts produced during planning. These texts include:

• Operations Plan

• Technical Proposal

• User Needs Requirements

• Project Management Plan

In my capacity as a technical writer for the Administration software, I used these texts to familiarize myself with the design intention, software specification, user requirements, and plan of the development schedule. Understanding of these texts allowed me to have an idea of what I was going to be writing about and who I was going to be writing for. They provided background information and a conceptual framework for my participation in the Administration software project. Because these texts signify and represent various milestones within the systems development process, they facilitate the process.

Relationship of texts to systems design

The texts that are analyzed in this chapter can be understood with respect to these phases of structured systems development. The following provides some background information on the software design process and the position of the texts within the structured systems development process.

Background on software design process

Both planning and development are a part of a software design process. R&D Tel. has developed its own software design process. This design process is considered proprietary, and will not be specifically discussed here. What follows is a general description of the stages of structured systems development.

Stages of structured systems development

The systems development process is characterized by three following stages: systems analysis, systems design, and implementation. The first stage, systems analysis, is a process of analysis where the scope of the project is defined. Traditionally, systems analysis consists of information-gathering and information analysis. The end result of the systems analysis stage is some kind of systems analysis report. Usually, the systems analysis report will include the following: a) a statement of the scope and objectives af the systems analysis; b) an explanation of the present system; c) a statement of all constraints on the system and any assumptions made by the analyst at this phase; d) a report on the (software and hardware) alternatives that seem feasible; e) an estimate of the resources and capital required. The systems design stage is traditionally characterized by the following: a) review of goals and objectives; b) development of a system model; c) evaluation of the organizational constraints; d) development of alternative designs; e) performance of a cost/benefit analysis; f) preparation of a system design report and recommendations. The implementation stage is traditionally characterized by the following: a) programming; b)

production of end-user documentation; c) training of users; d) conversion of the old system to the new. [10]

Position of texts in the system development process

The relationship of the texts in this chapter to the traditional systems development process follow. The *"Operations plan"* document was produced by R&D Tel. at the end of the systems analysis phase. The *"Technical Requirements"* document was produced by R&D Tel. at the end of the systems design phase. The *"User Needs Requirements"* document was produced by TelCorp. as the acceptance of the system design. The *"Project Management Plan"* was produced by R&D Tel. to manage the implementation phase.

Criteria of analyses

The criteria used in the analyses of the four texts in this chapter focus on the analytic foundation with respect to the system design process. The analytic foundation of the Administration software is based on traditional methods of computer science practice. These include: the context in which the systems analysis and systems design was conducted; the methodology of the analyses of the administrative processes; the assumptions made by the software designers during the analysis and design phases; and the use of background knowledge. These criteria will provide an analysis of how the view of the administrative processes was constructed during systems design.

Terms used in this chapter

In this case study, the investigation was conducted by R&D Tel. and funded by TelCorp. In traditional computer science, this is referred to as the Systems design phase. This investigation is also referred to as the "planning phase" for the purposes of this study. The main reason for this is that the people that conducted the investigation were members of a planning department at R&D Tel.

---

[10] S. Mandell, "Computers and Data Processing", pp. 319.

The term "user" refers to the population who will use the Administration software. These are members of the three families in TelCorp.

The term "user representatives" refers to members of the Methods departments who have been assigned to collaborate on the Administrative software project.

The term "user needs" refers to the requirements of the users and the user representatives in the Administration software.

Organization of this chapter

The organization of this chapter follows the chronology of texts produced during the systems design stage of the Administration software project. Background information about the Generic Forms Tool (GFT) is provided to appreciate how the concept of administrative processes was constructed in the Systems design phase.

The first analysis is based on the *"Operations plan"*. Analysis of this text reveals the primary objectives for initiating the project; the kind of methodology used in the system analysis stage; how information inefficiencies are captured through this methodology; and why a list of system features is recommended as a solution to the information problems revealed through this investigation.

The second analysis is based on the *"Technical proposal."* Analysis of this text reveals the kind of methodology used in the system design stage; how the administrative processes are constructed as the information flow between administrative units; and the technical view of the organizational future is described based on the assumption that all administrative processes are consolidated in the future.

The third analysis is based on the *"User Needs Requirements"* document. Analysis of this text reveals the kind of methodology used in analysis of the administrative activities at the end of the systems design stage and how this analysis provided a framework on which the background software could be mapped to the 'needs' of the current administration.

The final analysis is based on the *"Project Management Plan"*. Analysis of this text reveals that it was produced to bridge the systems design stage to the implementation stage; the project was broken down into many software tools to be managed; and the configuration of the Forms software had a management risk associated with it.

To summarize, the analytic framework is consolidated in analytic patterns. These patterns create a context in which *Chapter 3: talking about development* can be introduced.

Background knowledge

The Generic Forms Tool (GFT) software was produced at R&D Tel. before the Administration software project had begun. This software was created as a base software for the automation of the administration of another organization. R&D Tel. did not win this contract. However, it was left with this software which appeared to have great potential. In fact, it believed that this software could be used to automate the administration of any organization.

When I first started working at R&D Tel. there was a great push to market this software. In 1989, representatives from R&D Tel. presented the GFT software to a conference. The following is abstracted from that presentation. To begin, the rationale for creating this software follows:

> "A set of generic forms handling capabilities oriented to telecommunications operating company applications is presented. The approach provides for efficient electronic form processing, routing, and tracking with built-in flexibility to adapt to change. This is provided through a set of tools for on-line configuration of forms and processes, designed to be accessible by operating company personnel. Also discussed are networking requirements, including the need to interwork with existing administrative systems.

> Few applications are static. Most are affected by the introduction of new technology, organizational changes, or even new regulatory procedures. Many mechanized systems built over the last decade are costly to maintain and do not provide the flexibility needed to cope with such application volatility. The traditional development cycle (requirements, specifications, design, development, verification) is simply too inefficient to be responsive to requests for minor changes in information content and form...The need

exists for systems flexible enough to move responsibility for such minor (but necessary) changes closer to the user."[11]

The basic concepts of the forms software are described below.

> "The heart of the building block approach is a form profile which captures the description of the electronic form as well as the rules used to route form instances among users of the system...The profile also contains the permissions (e.g. field protection, special routing privileges) granted to users at various points in the process. A system administrator uses powerful system editors to modify form contents, layout, routing rules, and user permissions. The profile consists of three components: a template, a path, and permissions.
>
> The template describes the information contained on the form, as well as the physical layout...A *path* captures the process for routing form instances. It consists of a set of nodes linked serially or in parallel. Each node corresponds to a particular function which is to be performed. Associated with each node will be a user group responsible for this function. A *group* is a collection of users who typically share a common job function. A user can be a member of more than one group." [12]

The Routing Primitives are described as follows:

> "Routing refers to the passing of processing control among users. While the implementation may not require any actual movement of data, routing offers a handy metaphor, borrowed from the paper world, for both the user and the system administrator. A wide set of routing primitives is required to reconcile the competing objectives of: 1) maximal automation for highly-structured, well defined process flows; and 2) flexibility for process flows requiring decisions, as well as exception handling.

---

[11] Huberman, Tessler, Cameron, "A Generic Forms Handling Tool and its Application to Network Trouble Management", Proceedings from ICC '89, Boston, June 1989, pp.1.

[12] Ibid, pp. 3.

The routing primitives include the following:

> *Forward* This is the basic routing primitive. The form will be forwarded along to the next user group(s) designated in the pre-defined path.

> *Dispatch*: The dispatch command is a dynamic deviation from the current path. The user selects from a set of available "destination" paths...each node along the "destination" path has been configured with its own set of (field and routing) permissions.

> *Refer*: Refer also provides for a dynamic deviation from the predefined path. It is more of an ad-hoc deviation oriented to consultation either another user, or group, rather than a pre-defined path.

> *Close*: The close routing function is used to end the flow of a form, even in the middle of the path.

> *Sendback*: This operation permits a form to be returned to the sender, either to correct information, or because the form was incorrectly routed.

> *Sendcopy*: This operation does not affect the current process flow, but permits a snapshot of the current form to be sent to a user or group for information."[13]

The authors of this document admit that,

> "Setting up profiles is a non-trivial task for a system administrator, but programming expertise is not required. Rather, it is an understanding of the operational processes that is needed."[14]

In this sense, the configuration is associated with the task of constructing the (form) Profile. The tasks associated with constructing a profile include:

---

[13] Huberman, Tessler, Cameron, "A Generic Forms Handling Tool and its Application to Network Trouble Management", Proceedings from ICC '89, Boston, June 1989, pp. 3.

[14] Ibid, pp. 3.

"Profile will define the set of tasks to be performed on the data. If a form is a joint-use form, each distinct task will be represented as a node in the path of the form to be processed and routed. Predefined nodes in a path can have certain privileges and permissions. The template of each form will provide the mapping between the structure of information and its presentation. For example:

a) Procedures for automatic database extraction into the appropriate field name on the form.
b) Procedure of the final distribution of a form and the appropriate rules of distribution of the complete path."[15]

The GFT software is an electronic forms creation and distribution product. With it, a user can create forms and send them to other users who are connected to the forms network. There are three levels of privileges that have to be "configured". These privileges specify which users create forms, the routing functions (primitives) that users can access, and which other systems the users can access.

The GFT software was designed with the intention of bringing the customization of the software out of the hands of the software designers and into the hands of the user. The user responsible for the configuration of the GFT software would have to be a computer enthusiast. As the description of the tasks associated with the construction of the form profile imply, knowledge and facility with computers and systems theory is mandatory.

---

[15] Huberman, Tessler, Cameron, "A Generic Forms Handling Tool and its Application to Network Trouble Management", Proceedings from ICC '89, Boston, June 1989, pp. 3.

Operations Plan

The *"Operations Plan"* was written in 1986. The document was produced by a member of the R&D Tel. planning department, Don B. The plan was based on a thorough investigation of the administrative operations of TelCorp. During the investigation, Don B. went to TelCorp. and conducted a series of informal interviews, collected samples of administrative forms, administrative practices, user manuals from systems used in administration and talked to people involved in administering the telecommunications network. TelCorp. funded both this investigation and the production of the *"Operations plan"*.

The following provides a brief analysis of the objectives, methodology and assumptions contained in the plan, as well as the list of system features that make up the operations plan.

Objective

The objective of the project, at this stage, is phrased in the following way:

> To find a way to deploy the Thrust operations systems foundations into TelCorp. in the 1988-1989 timeframe in a way that would speed up the realization of strategic operations concepts.[16]

This objective can be understood on three separate levels of abstraction. First, the objective is to find a way of attaining strategic operations concepts. What these concepts are is left to the readers' imagination. The reader is expected to be familiar with these concepts. The only people likely to be privy to this knowledge are the upper management of R&D Tel. For the purpose of this analysis, however, it is suffice to say that the "Thrust operations" refer to a set of both systems planning and development projects that have been previously developed at R&D Tel. Furthermore, the "Thrust operations" have a common aim to standardize the practices and procedures of administrative operations. As a part of the

---

[16] Operations plan, pp.4.

"Thrust operations", this project represents a part of a strategic initiative of R&D Tel. The second objective, then, would be to dedicate resources towards the actualization of the "Thrust operations". The third objective is to identify areas of opportunity in software development that could be pursued in the following year.

Methodology

The methodology of the *"Operations Plan"* is based on the objectives of the "Thrust operations" to streamline the operations of the telecommunications network by making administrative information available in a computerized form and accessible by many departments. This objective is based on information that was collected during the Systems analysis phase.

The methodology is a standard flowcharting technique that is common to traditional computer science practices. This technique is based on a language for representing the structural and behavioral aspects of the telecommunications network through algorithms and decision making symbols.

Figure 1 illustrates some of the flowchart symbols used to analyze systems.

Figure 1

Sample flowchart symbols

Symbol	Description
Terminal interrupt	A, start, stop or interruption point in a program
Input/ Output	General input/output operations
Process	Any processing functions
Decision	A comparison operation that determines which of the two alternative paths is followed
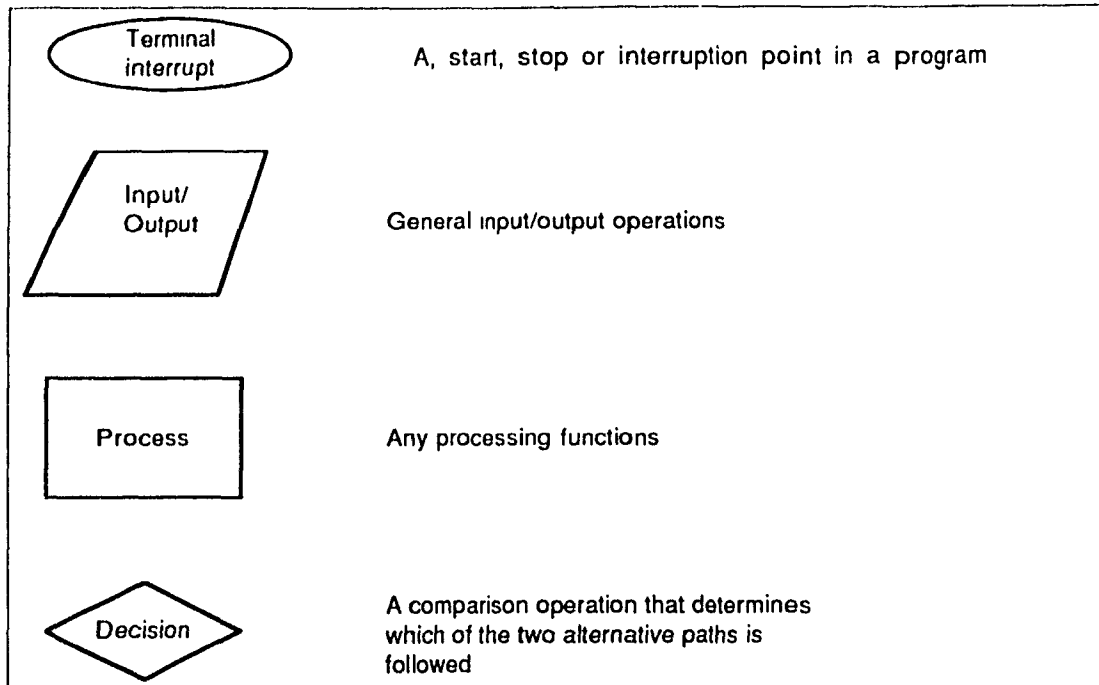
When flowchart symbols are connected with arrows, they indicate the flow of the information, the places where information is processed, the places where information is entered, and the places where decisions are made, etc. Figure 2 illustrates a sample flowchart for a grocery bill calculation.

Figure 2

Sample flowchart for a grocery bill calculation
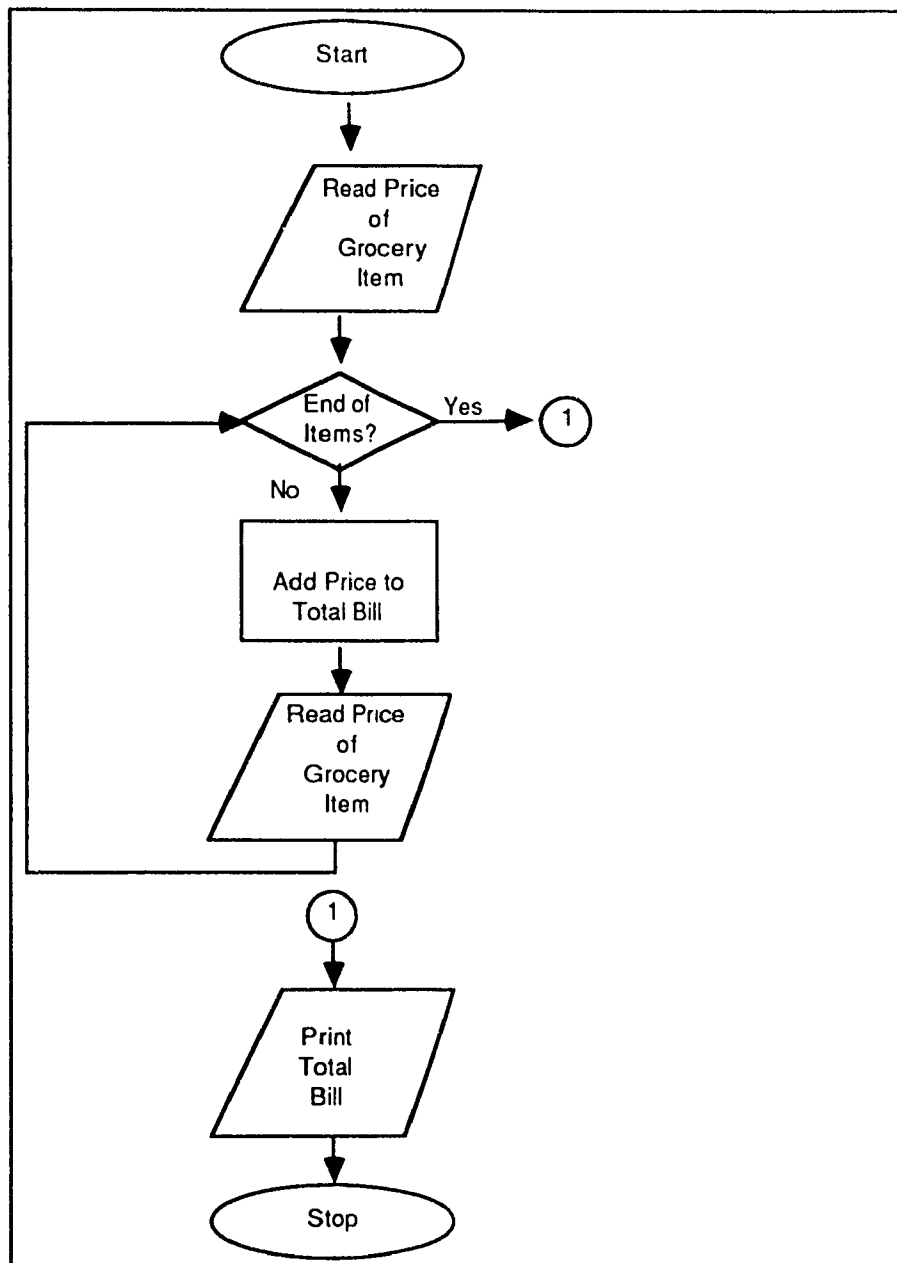
This is the type of diagram that was created to map out the administration of TelCorp. The

mapping of the TelCorp. administration resulted in eight functional categories. A function

is based on information processes (collection, download, modification of information).

These are paraphrased in the following:

1) Manual collection of information

2) Manual download of information to the network

3) Electronic download of information to the network

4) Record keeping of information with computer systems

5) Record keeping of network usage with computer systems

6) Audit systems

7) Procedural systems

TelCorp. administration is described as a set of *information processes* and decision-making rules. As such, the administrative processes are likened to information processes. It was possible to describe the administrative processes as information processes with the traditional formal flowcharting technique.

Assumptions

The methodology of the operations plan was based on the following assumptions:

1) The recommendations should be based on the Thrust operations systems foundations.

2) A characterization of TelCorp. administration, specifically administration that would support various TelCorp. departments.

3) Recommendations should be working "in the field" by the first quarter (IQ) of 1989.

4) Recommendations should have short-term economic benefit to TelCorp.

5) Recommendations should be consistent with strategic (Thrust' operations views.

Based on these assumptions, the operating plan frames the project. The following high-level organizational objectives of R&D Tel. are revealed: *technological*—to use a software tool that had been previously developed by R&D Tel. but had never been sold to a customer; *economic*—would be economically beneficial to R&D Tel.; and *strategic*—would accomplish the strategic objectives of the research company. The conception of the automated administration project was clearly bound by these strategic objectives of the research company.

List of system features

Based on the analysis of how best to deploy the "Thrust operations" within the boundaries established in the assumptions section, the *"Operations plan"* includes a section entitled "Recommendations for follow-up in 1987." The recommendations are a list of system features, as well as the benefits that relate to the feature.

The following features are relevant to this discussion:

> **Feature** - mechanization and automated update of the administrative databases.
> **Benefits** - elimination of clerks performing manual input during network administration, and of clerks performing manual updates to the maintenance of the network, and also elimination of associated input errors.

> **Feature** - automated validation of data consistency among databases and data residing in the network.
> **Benefits** - database synchronization reducing errors.

Known in the computer industry as a Feature Rollout, the recommendations are a list of system features. Each of these features covers a broad administrative area. In this sense, a feature relates to an administrative function. The benefits are based on increased information processing efficiency.

These recommendations are based on the analysis of the administrative processes as information processes. Only, when the administration was categorized in information processing categories could information processing features be recommended.

Summary

The investigation that resulted in the production of the *"Operations plan"* was the investigation that precluded the systems analysis stage. The analysis of this phase breaks the organization into information processing functions. Each function is determined by the information needs of the organization as a whole.

The objective of the systems design stage was to streamline the telecommunications operations. The methodology used to analyze the administrative operations was based on formal symbols for representing the structural and behavioral aspects of the telecommunications network. The flowcharting technique is a traditional computer science practice. This technique resulted in the functional analysis of the administration where each function represented a process to administrative information. Through this methodology, administrative processes were analogous to information processes. In this light, the recommendations for automation based on the same information processing categories is consistent.

Technical proposal

Once the *"Operations plan"* was accepted by the upper management of TelCorp., there was a need to define the technical specifications of the automated administration. These technical requirements are specified in the *"Technical proposal"*. Again, the authors of this text are members of an R&D Tel. planning department.

The following is an analysis of the *"Technical proposal"* and the concept of the administrative processes that is implicit in the technical proposal. Included are an analysis of the structure of the document; a description of the structured analysis methodology which was used in the production of the document; the operational benefits of the proposal; a description of the communicational function of information flow; the technical view of the organizational future that is articulated in terms of new service offerings and an acknowledgement of the relationship between the automated administration and the organizational structure.

Structure of document

The headings used in an official document of this sort represent, on a certain level, how the document is structured to be read, what the authors consider to be significant, as well as, the analytic framework of the investigation. The following is an analysis of the title and major headings of the technical proposal. The table of contents of the technical proposal follows:

The title indicates that the document is a technical proposal. What is this document proposing? From the table of contents, it appears that the document provides an analysis, a proposal, and a plan of action. Chapters one through five contain a scientific analysis of the current administrative procedures of the customer's organization. Chapters 6 offers the technical principles for Chapters seven through nine which proposes a new method of administering the customer's organization. Chapters ten through twelve provide further rationalization for the proposal. Chapter thirteen is a plan for implementing the proposal. The structure of the document emphasizes the analysis of the current administrative procedures and the proposed technique of administering the customer's organization.

The objective of the technical proposal, though, is to get funds to support software development at R&D Tel. Consequently, the analyses are based on hard facts. These hard

facts are presented in a tight analytical and scientific manner. For instance, the analysis of the administrative work is presented as Data Flow Diagrams. Similarly, the economic analysis is based on statistical data. The presentation of this information in this manner is rational and hard to argue with.

Methodology

The methodology used to create the technical proposal is evident in both the techniques used to gather information and the interpretation of this information.

Techniques used in information gathering

The techniques used to gather information about the administration of the customer's organization are: a) the study of practices; and b) a series of interviews.

The practices are administrative practices written by various methods groups. These practices are the official documentation for work procedures. There are "practices" for just about every task included in every job at TelCorp. There are two assumptions that the authors made about these practices. First, they assumed that the practices were current. Second, *they assumed that the practices represented how the work is actually performed.*

The interviews that were conducted were with members of the three families, or line groups. Samples of these interviews are not available for inspection. It is, therefore, impossible to determine what type of interviews were conducted (formal, semi-formal, etc). It is also impossible to tell what kinds of questions were asked during these interviews. The authors assumed that the samples of interviews were not important to include as research data.

Techniques used in information interpretation

The interpretation of the information is based on a "structured analysis" methodology. This technique is also a formal method of analysis. The structured analysis is based on a Data Flow Diagram (DFD) tool. The DFD are maps of data flow that are represented by arrows (signifying the direction of the data flowing) in between functions (represented by a box with rounded corners). A function, in this document, is qualified as a change, or modification, of one of the incoming data flows. The authors assume that when people work with information they are performing functions. An advantage to the view of activity as functions is paraphrased in the following:

> The ability to expand a function into a series of sub-functions is one of the attributes which make the DFD a powerful tool. If we have, for example, a DFD with four functions numbered 1 through 4 and if we decide to expand function number 1, we could create another DFD with function numbers 1.1, 1.2, 1.3 and so on. Like the layers of an onion, the DFD allows us to go down to the desired level of specification by breaking down its functions into these sub-functions.[17]

Figure 3 illustrates a sample Data Flow Diagram.

---

[17] Technical proposal, pp. 65.

Figure 3

Sample Data Flow Diagram
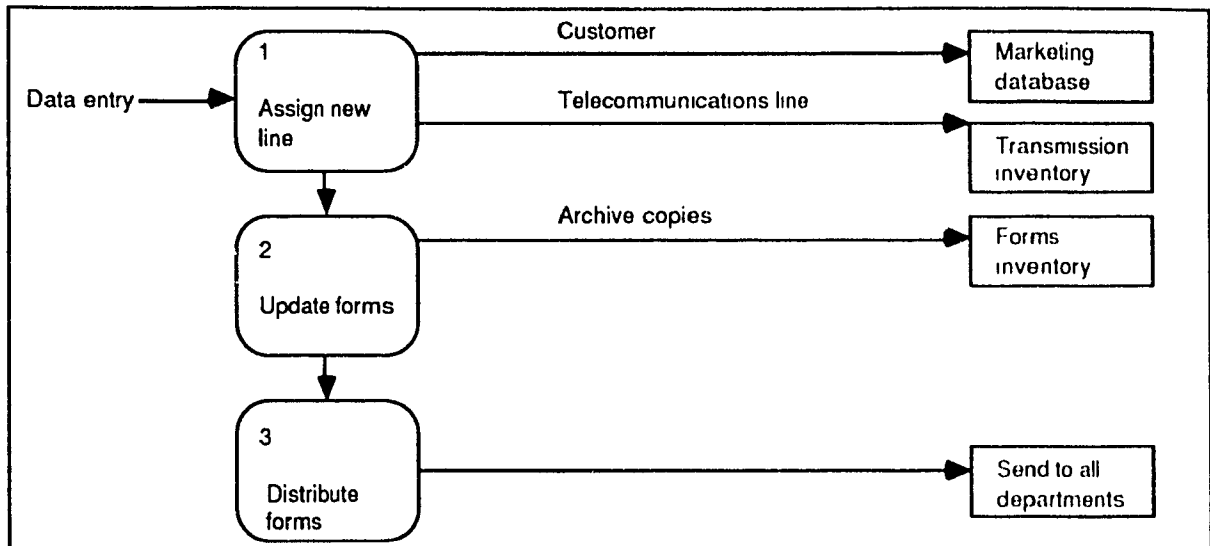
The purpose of this analysis is paraphrased below:

> Based exclusively on the data flow, we were able to identify problems due to *redundancy*, to *duplication*, to information present but *not required* and to *unnecessary manipulation*. When information enters a system characterized by a DFD, we can follow its path and transformations until it leaves the system. Functions are defined purely by changes brought to the information flow. *A DFD does not suggest any type of administrative structure, its goal is simply to capture concisely the flow of information.*[18]

Information flow can be captured apart from organizational structure.

---

[18] Technical proposal, pp. 7.

Symbolic view of organizational reality

Moreover, the concise flow of information is regarded, from this point-of-view as an accurate image of reality. The following is paraphrased:

> We conclude by offering the whole set of DFDs which describe the administrative process. We must mention that these DFDs are not 100% accurate; there are certainly a few omissions. Nevertheless, we believe that they still embody a sufficiently accurate representation of reality so as to form a solid basis for our proposal.[19]

Reality, then, is based on the formal representation of information flow which can be captured apart from organizational structure.

Communication and control issues related to current information flow

The *"Technical proposal"* includes a section titled "Conclusions Derived from the Analysis". These are outstanding problems that are worth mentioning but could not be represented by using the DFD technique. The following is paraphrased from this section:

> • The current framework surrounding administrative form preparation and, more specifically, the involvement of a large number of groups, accentuates the need to circulate paper (forms, memos, etc.) *to maintain a functional level of communication.*

> • The importance of *maintaining good communication*, especially in the case of exceptions, requires the creation of a great range of distinct types of forms. Many of these forms are only used in a single administrative center and are the *result of local initiative.*

> • The large number of groups and individuals involved in the process dilutes each individual group's responsibility. This complicates tracking administrative forms since their is *no single person or group responsible* for its completion.

---

[19] Technical proposal. pp. 70.

• The large number of groups also *limits the control* which the administration can have on the throughput of the process.

• The relatively slow pace of the process (requiring 46 days for a form) means that it is difficult to detect and solve errors, particularly if this solution requires the coordinated efforts of several of the groups involved in the process. *This difficulty stems from the fact that none of the groups has a truly global view of the process.*[20]

The assumptions highlighted in the section on conclusions follow. In the first point, the authors assume that communication in the organization is functionally based. Furthermore, the multi-group administrative process imposes paper-based communication. In the second point, they assert that many forms are created to maintain good communication. Based on the first point, it seems fair that *communication is functionally based.* Also, the authors add that some of these forms are the "result of *local initiative.* I understand this to mean that some forms are created for communication within one group, or department. The third point assumes that some person or group should be *responsible* for each form. The fourth point assumes that the administration (a.k.a. management) should have full *control* of the processing of forms. Finally, the authors conclude that nobody in the customer's organization has a *global view* of the procedures involved in administrative processes.

A common theme of these conclusions is that they relate to the "communicative fabric" of the organization. The fact that they are under this heading in the document implies that these are outstanding issues that need to be addressed. Communication between individuals and between groups is connected to administrative responsibility and control of the administrative process. The nature of this connection is unclear as a set of conclusions. However, the view of the organizational future seems to indicate how the authors believe these issues could be settled.

---

[20] Technical proposal, pp. 15.

Technical view of the organizational future

The technical view of the organizational future is that all employees will have access to a workstation and all administrative tasks will be accomplished from this work station:

> The first principle involves consolidating all steps in the assignment (administrative) process so that they may be completed *from a single workstation.* Ideally, it should be possible to have everything done *by a single clerk.* However, the proposed solution must be flexible enough with respect to task assignment so as to be workable even if the assignment (administrative) process is not consolidated as soon as the mechanized system is placed in service.[21]

The view of the future organization is that all employees involved in administration will have access to computer workstations. An inherent objective in this view is to consolidate administrative procedures to the point where a single employee could perform them. This is the view of the ideal organization that is inherent in the *"Technical proposal".*

In this view, the communicative needs of the current paper-based administration are eliminated. The workstation view is coupled with the belief that it is possible to consolidate all administrative practices. In this view, the functional requirements of communication would be unnecessary. For if all administrative practices are consolidated, a single clerk could perform them, and hence, no one would NEED to talk to anyone else. Also, if all administrative practices can be performed by a single clerk then individual responsibility for administration makes sense. Finally, if individual clerks can be responsible for complete administrative processes, then control over processes and clerks can be accomplished through supervision.

---

[21] Technical proposal, pp. 18.

This vision of the organizational future is also tied into new services that TelCorp. will offer in the future. The following is paraphrased from this text:

> In the past few years, R&D Tel. has attempted to promote Operations Architecture aimed at updating in a consistent way, the data processing tools used by TelCorp. for managing its telecommunications network. Several technical and administrative questions were identified. For example, what is the best way to develop operations to guarantee complete and consistent service to the Integrated Service Digital Network (ISDN) customer?... many technical and administrative recommendations were made to improve TelCorp. operations in *view of the anticipated new service offerings* whose development responsibility should *normally be distributed among many administrative groups.*[22]

The anticipated service offerings that the authors speak of refer to new services that the telecommunications company intends to offer to its customers. These services are consistent with the "ISDN customer". ISDN is a view of the telecommunications network that is wholly digital. This proposal, therefore, is written with a view of a digital telecommunications network. The view of the organization is that the administrative functions support a digital network which does not currently exist. Also, the administrative responsibility of these new services should normally (should read currently) be distributed among many departments. The assumption is that the administration of the new services will be effected across many departments involved in the administration of services to customers. In an organization where a single clerk can perform all administrative functions, the administrative changes that these new services imply could be focussed. If many departments are centralized than new service offerings will be limited to one "department".

---

[22] Technical proposal, pp. 1.

Acknowledgement of the relationship between the automated administration and organizational structure

The proposal also articulates some difficulties in dealing with potential changes in the structure of the organization. The following explains how a change in the technology of the administration may effect the administrative processes:

> It is impossible to make a proposal without considering the potential impact on the administrative structure of TelCorp. and it is up to TelCorp. to decide on the need for administrative restructuration. For this reason, even if the proposal suggests the idea of a single step for administration, it allows for the division of the administrative task between many people if such is the chosen administrative structure.[23]

The authors state that the view of the single-step administration will necessitate a restructuring of the organization. The structure of the administration, they conclude, is the responsibility of TelCorp. They absolve themselves of this responsibility. They assume that the TelCorp. management will deal with this issue as they see fit. This is the only instance in the document where the relationship between the administrative software and the administrative structure is mentioned.

Buried deep in the middle of the proposal are the following warnings:

> • The proposed administrative system implies a significant restructuration of TelCorp.'s administrative framework. However, the business case on which this project is based had as objective the implementation in TelCorp. of technologies developed or being developed in R&D Tel. in the shortest possible time. The dichotomy between the administrative and operational impact of this project and the time frame which the business case dictated for the project's completion may cause some problems.

---

[23] Technical proposal, pp. 19.

• The exact infrastructure necessary for the distribution of administrative information has yet to be determined. The solution we will propose will be as universal as possible.[24]

Summary

The techniques used in information gathering for the System design phase were based on the study of practices and interviews at TelCorp. During the information gathering process, the authors clearly assumed that the practices represented how the administrative work was actually performed.

The interpretation of the information gathered during the Systems design phase was based on a structured analysis methodology. This technique is a formal representation of information processing functions using data flow diagrams. The use of data flow diagrams illustrate a detailed analysis of the information flow between administrative units. In using this technique, the authors believe that administrative information flow can be captured apart from organizational structure.

The analysis also revealed some problems of the paper-based administrative system that could not be illustrated with data flow diagrams. These problems are related to communication and control in the administration. The paper-based administration is complex involving many groups and individuals. In the paper-based administration, individuals and groups have a need to maintain a functional level of communications. Furthermore, due to the complexity of the paper-based administration, it is difficult for individuals to take responsibility for administrative processes or for management to control these processes. Individual and inter-group communications make up for the absence of administrative responsibility and administrative control in the paper-based administration.

---

[24] Technical proposal, pp. 50.

The view of the organizational future has a technical basis. It is based on a workstation. This view assumes that all administrative practices have been standardized as software functions. Furthermore, all administrative information in centralized. In this view, the functional need for communications is eliminated. The view of the organizational future is also connected to new service offerings that are anticipated. As an afterthought, the authors acknowledge that the administration may have to be restructured if all administrative processes are consolidated in the future.

User Needs Requirements

Once the *"Technical Proposal"* was accepted by the upper management of TelCorp., further specification was required to match the technical requirements included in the *"Technical Proposal"* to the business requirements or 'user needs'. This was the opportunity for TelCorp. to express its own needs for an automated administrative system. These requirements would be the measures of acceptance for the System design of the automated administration system.

The *"User Needs Requirements"* document is written in 1988 by the TelCorp. Traffic Methods Manager, Jeff B. At this point, he has been assigned the prime responsibility of the development of the administrative system. The intention of the *"User Needs Requirements"* document is to specify the requirements of the system from the users' point of-view.

Included in this section are an analysis of the structure of the *"User Needs Requirements"*, and the background knowledge necessary for an accurate interpretation of the *"User Needs Requirements"*.

Structure of document

The headings used in an official document of this sort represent, on a certain level, how the document is structured to be read, what the authors consider to be significant, as well as the analytic framework of the investigation. The following is an analysis of the title and major headings of the *"User Needs Requirements"* document.

The title, *"User Needs Requirements"*, indicates that the document is a specification of the user needs requirements. What are the requirements? From the table of contents, it appears that it provides both general requirements and detailed requirements for

administrative forms and the database structure and functionalities. Chapters one through six *contain a definition of the business requirements of the current administration, a definition of the business requirements of the new administration, and a mapping of the new administration onto a logical data structure.* Chapters seven and eight specify objective of the performance and control of the system. Chapter nine specifies the criteria of the human interface of the system. Chapters ten through twelve specify strategies for the validation, acceptance and size of the system that the users require. Chapters thirteen through sixteen specify the support and training that the users will require before they can use the system. The structure of the *"User Needs Requirements"* document emphasizes the requirements of the new administration in terms of business functions and how these requirements could be mapped onto a logical (formal) data structure.

Methodology

The methodology used to create the *"User Needs Requirements"* is evident in both the background knowledge that the document was based on and the method of analysis used to interpret the administrative requirements.

Background knowledge

The *"User Needs Requirements"* was written with two types of background knowledge. The first is a comprehensive understanding of the administrative processes involved in the administration of the telecommunications company. The second type of background knowledge crucial to the creation of this document is a thorough understanding of the Generic Forms Tool (GFT) software that was developed at R&D Tel. The *"User Needs Requirements"* document could not have been written without these bases of knowledge.

Background on organizational processes

The authors of the *"User Needs Requirements"* document are members of the Traffic Methods department. As I have previously mentioned, the Traffic Methods department is responsible for the administrative practices and procedures of the Traffic family. Before transferring to the Methods department, Monique R. had worked in many positions in the Traffic family. In total, she has worked for twenty years in the Traffic family. She has worked in a multiplicity of positions over the years. Most significantly, each of these positions is affected by the automation of administration. This experience, then, enabled her to anticipate the kinds of changes to specific job functions that would result from automation. The work experience of Monique R. was indispensable to the planning of the electronic environment.

Background on the Generic Forms Tool (GFT)

By the time the *"User Needs Requirements"* document was written, the GFT software had been demonstrated many times for the Methods department at TelCorp. See *"Background knowledge"* in this chapter for a full description of the GFT software. Both Monique R and Jeff B. were very familiar with the functional capabilities of the GFT software.

Analysis of administrative activities

There are two levels of analysis in the *"User Needs Requirements"* document. First, the administrative processes are analyzed with a bubble diagram technique where each bubble represents a business (administrative) activity. Second, each bubble is exploded as a secondary bubble diagram where each bubble represents a feature of the Administration software.

The functional perspective provides the basis for the analysis of the user requirements in the *"User Needs Requirements"* document. In this analyses a function is qualified as a

change, or modification, of one of the incoming data flows. The authors assume that when people work with information they are performing functions.

Bubble diagram technique

The new system business function structure is illustrated in a bubble diagram. The bubble diagram consists of circles interconnected with arrows. Each circle represents a business function. Each circle is numbered and the name of the function is written inside. The arrows indicate the flow of information between functions. The bubble diagram that illustrates the business system requirements is star-shaped. A bubble number 0.0 is in the centre of the diagram. All other bubbles are interconnected and surround bubble 0.0. A sample bubble diagram is illustrated in Figure 4.

Figure 4
Sample bubble diagram
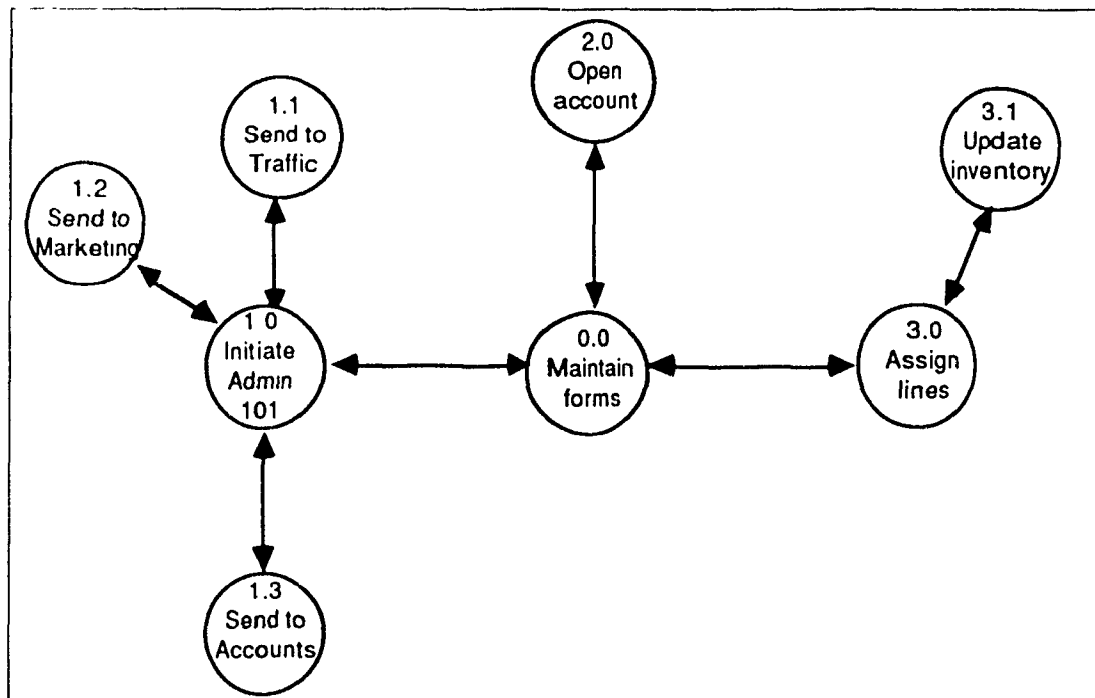
In the actual bubble diagram for the new business function structure of the TelCorp. administration, bubble 0.0. is labelled "Maintain Forms". From the users' perspective, this is the heart of the Administrative software. All administrative functions are connected to this function. The bubble diagrams illustrate an overview of the inter-relations of the administrative activities from a functional perspective.

Mapping of business functions to the Generic Forms Tool (GFT) functions

Each function included in the bubble diagram is referenced to another bubble diagram. The secondary bubble diagram is a function of the Administration software. This illustration is followed by a detailed description of the software function. Each description includes: the business activities a detailed specification of the function name, priority associated with it, description, associated features, estimated benefits and intangible benefits.

The crucial matter of this secondary analysis is that the Maintain Forms bubble (0.0) is referenced to the GFT software. The bubble diagram that is referenced to from the centre bubble, "Maintain Forms" is illustrated in Figure 5.

Figure 5

Secondary bubble diagram for Maintain Forms bubble
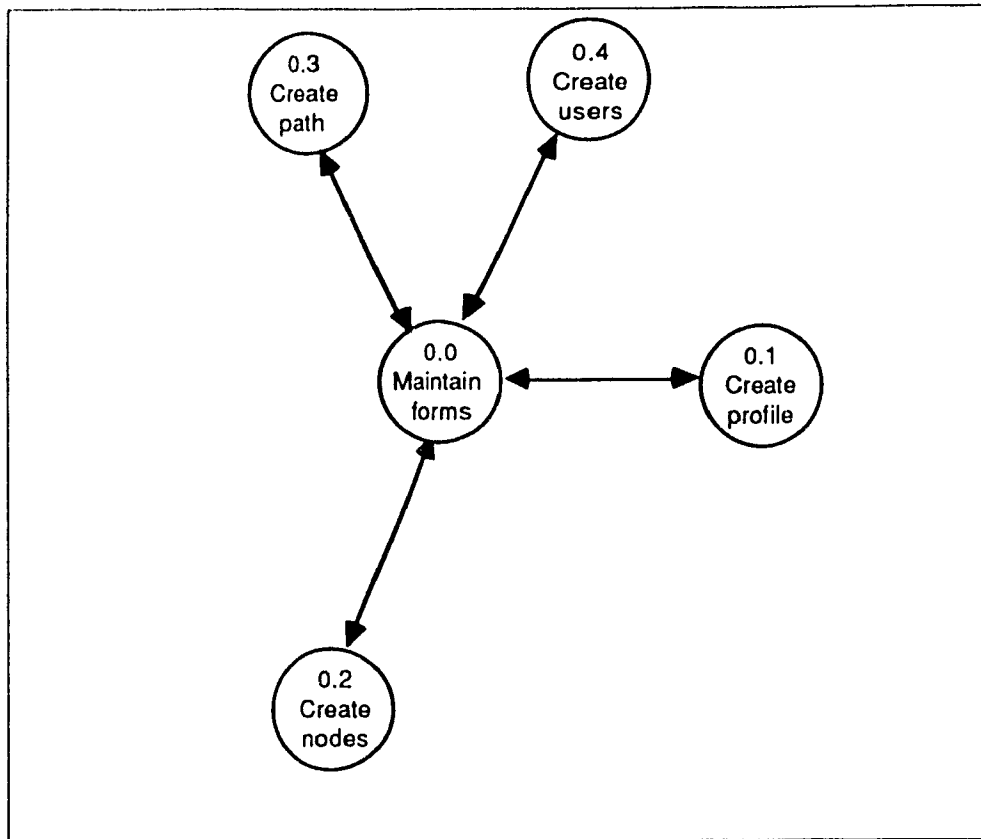
The detailed description of the "Maintain Forms" function is phrased in the following way.

> FUNCTION NAME: Maintain Forms (Bubble 0.0 in Business Functions
> Data Flow Diagram)
> PRIORITY: High
> FUNCTION DESCRIPTION:
> This function is basically required to perform the following:
> — Allow the users to create and update forms.
> — Allow the users to create and update path (i.e logical sequence of users in
> which the form is to be "routed" or become accessible).
> — Maintain all the forms and orders and either make them accessible
> or "route" them according to the established paths.

The secondary analysis was actually performed based on the functionality of the GFT

software. For the first time, the GFT software is introduced as the Forms software.

An appendix is included in the *"User Needs Requirements"* document. This appendix is a detailed definition of the Forms software. It is composed of the following sections:

1- Major requirement for forms software

2- *Node* processing

3- *Routing primitives*

4- Routing and processing forms

Suffice to say, that the authors assume some background knowledge in computer science, computer networks, and GFT in particular to understand these detailed specifications (See *"Background knowledge"* in this chapter for more information on GFT).

Knowledge of the functions of the GFT software provided an instrumental framework in which to categorize the administrative processes. The mapping of the administrative functions to the GFT software gave an instrumental framework to the administrative processes. The concept of the administrative processes was transformed into an instrumental concept at this time. In this case, the formal analysis of administrative processes were attached to an instrumental set of software features.

Summary of methodology

The techniques used to analyze the user need requirements as specified in the *"User Needs Requirements"* document were based on the following:

a) functional breakdown of business activities

b) mapping of business functions to GFT functions

c) detailed definition of Maintain Forms function

Each of these techniques required a functional analysis. The interrelationships between these techniques are: The detailed specification of business activities (b) is a functional analysis of the same type as the functional breakdown of business activities (a). In fact, technique number two is an extension of technique number 1. Both are based on the analysis of business functions. The first is the function in the business context. The second is a function within the context of the Forms software.

The mapping of business activity functions and onto the pre-designed GFT software functions is crucial to understanding how the functional analyses of the business activities were associated with the Forms software. The Forms software functions were created before the new business activities functions, as GFT software. It is reasonable to conclude, therefore, that the business activities functions were designed to accommodate the structure of the Forms software functions.

Summary

Background knowledge of the administrative processes were based on the individual work experience of Monique R. Her particular work history of organizational practices, procedures and contacts was the main source of the current administration of the organization. Additionally, the Traffic Methods people (Jeff B. and Monique R.) mastered the use of the GFT software. The terminology, capacities of the administrative system, and the 'look and feel' of the users' needs were founded on this software. Furthermore, the analysis of the business requirements of the users need were conceptualized so that they would coincide with the design of the GFT software.

There are two levels of analysis in the *"User Needs Requirements"* document. In both cases, a bubble diagram technique is used. On the first level, the administrative processes are analyzed with a bubble diagram technique where each bubble represents a business

(administrative) activity. This first level of analysis was a logical data structure. On the second level, each bubble is exploded as a secondary bubble diagram where each bubble represents a feature of the Administration software. The first level of analysis provided the conceptual framework in which the background software could be mapped to the 'needs' of the current administration. In this case, it is of specific interest that the centre bubble, which represents the main software function of the Administration software, is a detailed definition of the GFT software.

Product Management Plan

Once the *"User Needs Requirements"* document was negotiated and agreed upon between TelCorp. and R&D Tel., a contract was negotiated for the design and implementation of the Administration software. This was the end of the System design phase.

In 1989, the project was transferred from the planning department to a software development group at R&D Tel.. This was the beginning of the Implementation phase. The Manager of this group, Stewart D., authored a Project Management Plan. The purpose of the *"Project Management Plan"* was to specify how the Administration software project would be managed. The first Project Management Plan was issued in June 1989.

Through the development cycle of the following year, the management of the project was re-evaluated. Another Project Management Plan was issued during that year. The following is paraphrase from the second issue of the *"Project Management Plan"* :

> The Administration software was originally planned in three releases. 1.0, 2.0 and 3.0. It was realized after the first cycle of development that a great deal of software was required for the conversion effort. Instead of developing and using some ad-hoc software, it was decided to implement full-featured conversion tools which would make up a part of the Administration software 1.0. The milestone for the first software cycle was passed in May of 1989. It did not detail the Administration software 0.1. It focused on Administration software 1.0. As such, much of the documentation for the 0.1 release was incomplete. This project management plan has been re-written to take the 0.1 release into account.

Analysis of the Project Management Plan

The second *"Project Management Plan"* was written in 1990. At this time, the manager was newly appointed. The intention of this document was to specify how the Administration software project would be managed. Included in this section are an analysis

of the objective of the document, the structure of the document, background knowledge about the creation of the document, the view of the project included in the document, the methodology used in the production of the document, the assumptions included in the document, and the risks associated with the project.

Objectives

There are two objectives of this document. The first is to document the development that had occurred during the previous year. As Stewart D. said, the software development that occurred during 1989 occurred rapidly and, as a result, was mostly undocumented. The second objective of this document was to define the scheduling of activities, identify the management and operational structure that would be responsible for the product development process.

Structure

The headings used in an official document of this sort represent, on a certain level, how the document is structured to be read, what the authors consider to be significant, as well as, the analytic framework of the plan. The following is an analysis of the title and major headings of the Project Management Plan. The table of contents follows:

Chapter 1: Introduction
Chapter 2: Project Deliverables
Chapter 3: Project Organization
Chapter 4: Managerial Process
Chapter 5: Technical Process

The title indicates that the document is a plan for the management of the software project. What is to be managed? From the table of contents, it appears that the document specifies the project deliverables (refers to specific software components to be completed at a certain point in time); specifies how the project is compartmentalized, specifies the people assigned

to develop the software and people assigned to successfully implement it in TelCorp.; and specifies what the managerial techniques and the technical processes that will support the project.

View of the project

The Project Management Plan outlines the view of the project from the perspective of development. There are two primary indicators of what this perspective is. The view of the Administration software project is illustrated in the structure of development and the assumptions contained in the production of the Project Development Plan as described by the authors.

Structure of development

The development perspective specifies the components of software that have been identified to be produced.

The structure of development is depicted in Table 1.

Table 1

Structure of development

Phase	Administration software releases	Features	Deployment Date
I	Administration 0.1	Conversion tool 1 Conversion tool 2 Conversion tool 3 Conversion tool 4	IIIQ '90
II	Administration 1.0 Administration 1.1	Admin tool 5a (Access 1) Admin tool 5b (Access 2) **Admin tool 6 (Forms software)** Admin tool 7 (Query) Admin tool 8 (Transmission assignment) Admin tool 9 (Circuit design) Admin tool 10 (Traffic assignment)	IIQ '91
III	Administration 2.0 Administration 2.1	Automatic assignment Networking Admin	IV '91
IV	Administration 3.0	Download (to other administering databases)	IIQ '92

Table 1 illustrates how development of the Administration software is categorized. The columns, from left to right, include the following: the phases of the software development; the software releases that relate to the phases; the software features that are included in each software release; and the date of completion of each phase.

The view of the Administration software, from a development perspective acknowledges that the conversion of existing paper records is a distinct project on its own. This realization is illustrated in the distinction of the Administration software project into Phases I and Phase II. This distinction did not exist in the first Project Management Plan.

This view of the Administration software breaks the development into tools. The software features included in Phase I (Administration 0.1) are tools required for the conversion of paper records into an electronic format. Each tool is used to transcribe information into a

database. The Administration database is organized into logical types of information. A tool was developed for each logical type of information.

The software features included in Phase II (Administration 1.0 and 1.1) are tools required for the primary operations of the administration. These include accessing existing administrative records (Access), creating new records (Forms software), querying existing and new administrative records (Query), and assigning equipment to the telecommunications network (Assign). The Forms software is highlighted to show that this tool was considered to be one of many in the construction of the Administration software.

The software features in Phase III (Administration 2.0 and 2.1) are tools required to further automate the administrative process (Automatic Assign) and to permit the networking of the administration (Networking Admin).

Finally, the software features included in Phase IV (Administration 3) are intended to complete the automation of administration by electronically transferring administrative information to other administration databases.

The development of the project is broken down into tools for two reasons. First, each tool is assigned to individuals in the development group. This facilitates the management of the project. Individual developers take responsibility for the production of a complete feature of the Administration software. This form of management is a division of labour and complements the production of software Second, the sequencing of the development of features into phases is dictated by how the information is organized in the context of the organization's administration. This is to say that, for instance, existing information about

63

the telecommunications network is crucial to add new pieces of equipment to it. New administrative decisions are based on the content of the existing administrative information. Tools, then, had to be developed first that would meet the requirements of constructing the existing administrative records in a way that would enable the automation of new administrative records.

Assumptions

The following is paraphrased as it appears under the heading "Assumptions, Dependencies and Constraints":

> Form process and configuration will be performed by TelCorp operations primes. This is a non-trivial task for processes as complex as (administration) with many different organizational entities involved. *It is assumed that sufficient resources will be assigned at TelCorp to ensure alignment of operational procedures with the Administration software functionality, promote standards across regions and to configure forms and processes* (emphasis added).[25]

The view of setting up the forms processes is complex (non-trivial). The assumption here is that if enough resources (people and money) are working on this task it will be accomplished. It is clear that the authors believe that operational procedures can be standardized across regions. It is also evident that standardization of operational procedures must be complete in order for the Administration software to be fully functional. It is not so clear, however, whether the authors believe this task is an assumption, a dependency or a constraint.

---

[25] Project Management Plan, pp. 23.

Risks

As if to emphasize the same point under a different heading, the following is paraphrased
from the next sub-section called "Risk Management":

- **Form Process and Configuration**

    R&D Tel. will perform the initial form configuration. R&D Tel. will
    provide the necessary training to regional coordinators and support for this
    ongoing activity.

This time though, the activities associated with forms set-up of the forms software is purely
technical. Again, it is not clear why the form process and configuration is a risk to the
project.

Summary

The *"Project Management Plan"* depicts a view of the Administration software project from
the developers perspective. The structure of the document indicates that the project is
broken up into parts that support the software development managerial processes . To fully
appreciate the production of the *"Project Management Plan"*, the readers of this document
should know that it was written after a year of software development. The first Project
Management Plan was no longer meaningful for the software developed during 1989, was
far more complex than had previously been anticipated.

The view of the Administration software from the software developer's perspective is
based on two fundamental criteria. First, the development of the project is broken down
into tools to facilitate the management of the project. Individual developers take
responsibility for the production of a complete feature of the Administration software.
Second, the sequencing of the development of features into phases is dictated by how the
information is organized in the context of the organization's administration.

For the Administration software to be fully functional, the developers assume that if enough resources are dedicated to the configuration of the Forms software (including the standardization of operational procedures across regions), the Forms software will be functional on time. Finally, the developers foresee that if the technical aspects of the configuration of the Forms software are not completed it could risk the development of the Administration software.

Analytic patterns

The systems design phase can be characterized by patterns in the methodologies, dominant paradigm, assumptions, and background knowledge. These patterns are common to traditional practices of computer science. They are, therefore, the analytic principles from which the construct of the Administration processes was founded. The principles consists of the following essential elements:

- –constructed in isolation

––formal analysis of administration

–––a functional interpretation of administrative processes which were equated with

      information processes

–– mapping of administrative processes to pre-existing software

–––assumptions about standardized administrative practices

The following summarizes the analytic patterns on which the concept of the Administration processes were constructed.

Social context of the production of texts

The Systems design phase was conducted in isolation with very limited interaction between R&D Tel. and TelCorp. The *"Operations Plan"* and the *"Technical Proposal"* were produced by R&D Tel. R&D Tel. gathered information from TelCorp., yet, wrote the documents on their own. Afterall, R&D Tel. was funded to conduct the investigation that lead to the production of these texts. The *"User Needs Requirements"* document was signed by the user representatives of TelCorp. yet, the basis of the users' needs was founded on the GFT software, which was created at R&D Tel. Finally, the *"Project Management Plan"* was, once again, produced in the isolation of the laboratory of R&D Tel. The interaction between the software planners and the user representatives was client-based. The end result is that the texts produced during the systems design phase was isolated and, therefore, detached from the context of TelCorp.

Formal methodologies

R&D Tel. defined TelCorp. as a collection of inter-related administrative functions through the use of a series of formal methodologies. Each of these techniques was based on a formal representation of the administrative processes. A block diagram technique was used in the *"Operations Plan"* to map out the information processes and decision-making rules to be established for the organization. A Data Flow Diagram technique was used in the *"Technical Proposal"* to further specify the information problems "due to redundancy, to duplication, to information present but not required and to unnecessary manipulation". Finally, the bubble diagram technique was used on two levels of analysis in the *"User Needs Requirements"* document.

Functional paradigm

Common to the practices of traditional computer science, the formal analyses of administrative processes were conducted within the functional paradigm Functions were defined as changes to information flow.

The formal techniques were used to analyze the administration into administrative units Each unit was further analyzed into functional tasks that are performed through administrative practices. Then, the flow of information was mapped between these administrative units. Each time information changed states, or was modified, this change was highlighted. By using three different levels of functional analysis, the administration was analyzed into information-processing states within the administration.

By using these formal techniques on analysis on the functional paradigm, R&D Tel. was able to show that there was a duplication of administration records and, in some cases, administrative functions within the organization. This was a successful argument in

convincing TelCorp. that it could be more efficient by eliminating the redundancy inherent in the current administrative processes.

The functional framework was also used to create a plan to manage the development of the Administrative software. In this framework, the Administration software was analyzed into individual software tools. Software tools were assigned to individual software designers. The software tools were organized based on the information requirements of the conversion process where the administrative paper records had to be converted to an electronic format.

In this case, the functional paradigm also provided the framework for the administrative functions to be mapped directly on the GFT software.

Background knowledge

Background knowledge of the GFT software enabled the administrative functions to be mapped on to the pre-existing software. The GFT software had been demonstrated for members of TelCorp. on numerous occasions. They were familiar with the functions that it could perform. Knowledge of the functions of the GFT software provided an instrumental framework in which to categorize the administrative processes. The mapping of the administrative functions to the GFT software gave an instrumental framework to the administrative processes. The concept of the administrative processes was transformed into an instrumental concept at this time. In this case, the formal analysis of administrative processes were attached to an instrumental set of software features. This was where the systems design was accepted by TelCorp.

Assumptions of Systems design

In traditional Systems design, there is room for system designers to make assumptions about the social context of the organization. Because these assumptions cannot be formalized, they were not included in their analysis.

In this case, the analysis of the texts in this chapter reveals a static characterization of the organization through the assumption in the Systems design phase. These assumptions include: a) the written administrative practices that were used in analysis were accurate representations of how the users work and b) the organizational structure implies homogeneous job responsibilities in different regions c) inter-departmental relations are consistent d) administrative practices can be standardized and e) the use of administrative information is predictable across the organization. These assumptions constructed a static view of the organization.

Conclusion

The analytic foundation of the design of the administrative processes embodied five primary techniques of constructing and maintaining a view of organizational reality based on traditional computer science. First, common to traditional practices of systems design in computer science, it was conducted in isolation and was, therefore, detached from the context of TelCorp. Second, through the traditional practices of systems analysis, a formal view of the organization was constructed with formal techniques of analysis. Third, based on traditional practices of computer science the formal analysis of the administration was interpreted in a functional paradigm where administrative processes were defined as information processes. Fourth, within the functional paradigm, the administrative processes were mapped onto an instrumental base (existing software). This approach transformed the formal view of the organization into an instrumental view of the administrative processes. Finally, in the traditional approach to systems design there is ample opportunity to make assumptions about the social dynamics of the organization (ie. variance of administrative practices, growth of organizational structure, variance of job responsibilities). In this case, these assumptions were based on a static conception of the organization. In summary, in the systems analysis and design stages, a concept of administrative processes was constructed that embodied the following characteristics: detached, formal, had an instrumental base and operated within a static view of the organization (administrative practices, organizational structure, and job responsibilities).

Vision of organizational future

The analytic patterns are encapsulated in the view of the organizational future. This is a technical view. In this view, all employees have access to workstations. Each workstation is equipped with a version of the Administration software. The administration information is centralized in logical, corporate-wide databases. All administrative tasks can be performed with the Administration software from any workstation, regardless of the

71

location or department. In this view, administrative processes are routine, based on standard administrative processes that are homogeneous across the organization, and can be performed by anyone, regardless of his or her administrative experience.

This chapter has described how the administrative processes were conceptualized during the planning of the Administration software. This is intended to provide a framework for *Chapter 3: talking through development,* where I will show how the concept of the administrative processes changed through conversation.

Chapter 3: Talking through development

Introduction

The intention of this thesis is to show that the conceptualization of the administrative processes changed from the planning context to the development context. To show how this change occurred, I will compare how this view of the administrative processes was modified to incorporate the social dynamics of the organization through conversation during implementation.

Intention of this chapter

The intention of this chapter is to examine how the social dynamics of the organization were accommodated in the concept of administrative processes through the software development process. The way that the social dynamics of the organization were incorporated into the concept of the administrative processes was through conversation between the software designers and the user representatives. The social dynamics of the organization include: the organizational structure; the support structure of the organization; the administrative activities; the administrative practices; the use of information, and the user community.

The view of administrative processes was reformulated during conversation between the software designers and the user representatives. The concept and subsequent construct of the administrative processes was reformulated through conversations that dealt with the social dynamics of the organization. Through conversation, the social dynamics of the organization were incorporated into the construct of administrative processes in the Administrative software.

Context of conversations

With respect to the Administrative processes, the context of these conversations can be understood within the historical context of the analytic foundation as discussed in *Chapter 2: analytic foundation*. These texts also provide the conceptual framework for the conversations that are contained in this chapter. This framework can be described as a view of administration processes which was constructed that embodied the following characteristics: detached, formal, having an instrumental base and operating within a static view of the organization (administrative practices, organizational structure, and job responsibilities).

Chronicle of conversations

I was employed as a technical writer at R&D Tel. from 1988 through 1991. During this time I was personally involved in this project. Moreover, I was assigned to document many components of the Administration software. This experience drew me into conversations about the development of this software from many angles. My involvement spanned most of the software modules that were created as a part of the Administration software from 1990 throughout 1991. I have drawn from the participation in this project in the chronicling of this chapter.

The approach that I will take in this chapter is to show the impact of conversation on the concept of administrative processes during the implementation stage, is to document conversations that were held between various project members. These conversations took place between December '90 and July '91. The research data is presented as highlights from conversations that occurred. Relevant organizational texts are also included as research data.

The Implementation stage

For the purpose of this thesis, the Administration software project has been divided into two stages: the planning stage and the implementation phase. The planning stage was described through the analysis the texts in *Chapter 2: analytic foundation* . The implementation stage is described in this chapter.

The *"Project Management Plan"* for the Administration software revealed that the transition from a paper-based administration to an automated administration would be a long, and sometimes painful, one To begin with, all of the records that were maintained in a paper format would have to be transferred to a standardized electronic format that could be accessed from all types of terminals. Furthermore, these records would have to be organized in a logical database so that all information could be accessed when it was required. The integrity of the telecommunications network was based on the maintenance of these records. The automated administration, then, would operate on the maintenance of the same information.

The requirements for converting the paper-based administration to an automated one for over a thousand end-users were numerous, varied and complex. Due to the complexity of the Administration software, it is necessary to bound the scope of this research. Hence, the focus of this research is delimited to the Forms software.

The Forms software is the top layer of the Administration software. This is the software that is closest to the user. Consequently, it was this software that had to accommodate the needs of the user. This focus, then, provides an opportunity to closely examine how the concept of the administrative processes changed to incorporate the social dynamics of the organization at the heart of the social fabric of the organization, the user.

Implementation of the Forms software

As described in *Chapter 2: analytic foundation*, the Forms software was originally the Generic Forms Tool (GFT) software. There were additional requirements of the Administration software to access other customized software tools that were a part of the Administration software. (See the analysis of the "Project Management Plan" in *Chapter 2 analytic foundation* for a list of the software features that were included in the Administration software.) The intention of the Administration software was to provide access to the other software tools through the Forms software. For the user to access the auxiliary software tools through the Forms software, additional software development was required to modify the GFT software. The GFT software evolved into the Forms software as a result of these modifications.

Communication breakdowns

The technical complexities of the implementation of the Forms software were not the only complicating features of the project. As the project moved from the planning to the development phase, there were surprises and uncertainties at each step. After two years of being involved in the project, and a great deal of reflection, I can now say that much of the uncertainty was a result of communication breakdowns and misunderstandings between the software planners, the software designers and the user representatives. To illustrate how the communication breakdowns occurred during the development of the Administration software, I will follow the evolution of the language and concepts about the administrative processes that were conceived, re-evaluated, and reinterpreted as the Forms software development progressed. The language that was used through the construct of the concept of administrative processes was grounded in the software analysis and design context as I have shown in *Chapter 2: analytic foundation*. When the concept of the administrative processes moved from the planning phase to the development phase it caused confusion and misunderstanding between the software developers and the user

representatives. As an indexical expression, 'administrative processes' was no longer meaningful.

Indexicality

In *"On Formal Structures of Practical Actions"* Garfinkel & Sacks present a compendium of philosophical investigations of indexicals. Of particular interest to this essay are the thoughts on indexicals from the following:

> "Edmund Husserl spoke of expressions whose sense cannot be decided by an auditor without his necessarily knowing or assuming something about the biography and purposes of the user of the expression, the circumstances of the utterance, the previous course of discourse, or the particular relationship of actual or potential interaction between the user and the auditor...Bertrand Russel pointed out that descriptions involving them apply on each occasion of use to only one thing, but to different things on different occasions... Nelson Goodman wrote that each of their utterances constitutes a word and refers to a certain person , time, or place but names something not named by some replica of the word and also that ...indexical expressions and statements containing them are not freely repeatable in a given discourse in that not all their replicas are also translations of them."[26]

For the purpose of this essay, indexicality refers to the influence of a setting on a term, or an expression. It follows that a change of setting, or situation, will affect the indexicality of a term or an expression. In this case, the change of setting between the systems analysis and design stages, or planning phase, to the implementation stage or development phase, affected administrative processes. Each aspect of indexicality as cited by Garfinkel and Sacks is illustrates the change in the concept of the administrative processes between settings. They are summarized in the following:

---

[26] Harold Garfinkel and Harvey Sacks, On Formal Structures of Practical Actions, pp. 348.

First, the "biography and purposes of the user of the expression" has been well documented in *Chapter 2: analytic foundation*. The authors of the planning texts were computer scientists whose purpose was to define administrative processes so that they could be automated.

Second, as the implementation stage progressed, it became quite clear that "descriptions involving them apply on each occasion of use to only one thing, but to different things on different occasions". In the implementation stage, it became evident that administrative processes described not only changes to information states, but also described aspects of the social dynamics of the organization.

Third, as the software development of the implementation stage evolved, it became clear that the concept of the administrative processes was not literally translatable from the planning stage: "indexical expressions and statements containing them are not freely repeatable in a given discourse in that not all their replicas are also translations of them"

Fourth, the reference to the method of organizing the administration changed between the planning and development stages. As Nelson Goodman has pointed out, "each of their utterances constitutes a word and refers to a certain person , time, or place but names something not named by some replica of the word". What was unnamed in this case was the method of organizing the administration of TelCorp. During the development stage, it became clear that the method of organizing the paper-based administration would be different than the method of organizing the administration in an automated environment.

Finally, the referents to administrative processes changed dramatically between these stages. During planning, administrative processes were equated with information

processes. During implementation, it became quite clear that administrative processes were greatly influenced by the social dynamics of the organization.

Once the terminology and construct of the administrative processes were applied to the software development process, it became obvious that further understanding of the social dynamics of the organization had to be specified before the software could be implemented. Since the concept of the administrative processes had been constructed without any reference at all to the social context, organizational structure, organizational support structure, administrative activities; administrative practices, use of information, or the user community, there was a great deal that had to be addressed before the software could be developed in a way that would accommodate TelCorp.

In the context of planning, the term "administrative processes" was used both as a literal reference and as a construct in the design of the software. As a literal reference, the user representatives could not relate to the terminology associated with the administrative processes. As a construct, user representatives could not relate to how the concept was actually inscribed in the software code. For the inscription of the concept of administrative processes into software was evident throughout the operations of the Administrative software. The inscription of the concept of administrative processes effected how the major functions of the software were defined, the conditions under which functions could be performed, and the construction of the interrelations between functions.

The terminology and construct of the administration as it was conceptualized in the administrative processes were derived from the GFT software. Implicit in this software were terms and assumptions about the administration that did not match their representation in the paper-based administration.

Implementation of users on Forms software

The configuration of the Forms software involved the reinterpretation of the terminology and constructs of the administration as they were conceptualized in the administrative processes. The reinterpretation involved adapting the terms to the needs and social realities of the organization and to change the approach towards the mapping of the existing paper based administration to an automated one.

In the configuration of the Forms software, the inscription of the concept of administrative processes was inadequate with respect to the following specifications: a) creation of user groups; b) membership of user groups; c) permissions of users within user groups; d) relationships between user groups; and, e) sequence of processing of Administration form 101 to user groups. Each of these specifications about the users had to be determined to fully specify the administrative processes. Since these specifications had not been taken into account during the systems design stage, they had to be dealt with during implementation. These specifications were essential to the construct of the administrative processes in the configuration of the Forms software.

These specifications involved a reinterpretation of the concept of administrative processes This was where the administrative processes, as a term and as a concept of administration, could not be translated, but had to be reinterpreted. The reinterpretation of the concept of administrative processes, was most evident in the interactions and subsequent conversation between the software designers and the user representatives about the configuration of the Forms software.

Impact of conversation on software development

The impact of conversation in the development process can be attributed to how the interactions between R&D Tel. and TelCorp. were structured. TelCorp. assigned a democratic role to the user representatives in this project A user representative was selected from each of the three main families. Each user representative was supposed to represent its own family members. The role of the user representatives was to represent their family members in the software development process. In this role, they had to gain the approval of the development of the software from their users. They collected feedback from users at each step of the software development process.

For instance, to configure the Forms software, they went to the users who discussed every single field on the Administration form 101. There were over seventy fields on this form. Once the user representatives had collected feedback from their specific families, they would come back and discuss this feedback amongst themselves. At least that was how it was supposed to work. This was not how it worked in reality. Each of the user representatives had different interests in the project. They had to develop a way of collaborating that suited the situation.

Nonetheless, there was a great deal of talking that went on about the Forms software. User representatives did a lot of talking with their users, among themselves, and with the software designers. In my capacity as a technical writer on this project, however, I was privy to conversations mostly between the user representatives and the software designers, together and individually. The software designers spoke mostly with the user representatives. Conversation was constant in the software development process.

Social knowledge

Through conversation during software development, significant areas of social knowledge were revealed about the administrative processes and the implementation of an automated system. These areas of social knowledge had been either ignored or missed during the systems analysis and systems design stages. Through conversation, the following areas of social knowledge were revealed: knowledge of the structure of the organization; knowledge of the support structure of the organization; knowledge of administrative activities, knowledge of the use of information within the organization; knowledge of the user community of the organization. Through conversation, these areas of knowledge were specified as social facts that could be implemented into the Forms software. For the purpose of this essay, once an area of social knowledge was understood well enough to be accommodated into the development of the Forms software, it is referred to as a social fact

These social facts about the administrative processes were eventually incorporated into the design and/or the implementation strategy of the software. Once these knowledge bases were understood, the following social facts were accommodated in the design and implementation of the Administration software: support for a dynamic organizational structure; support for variable inter-departmental relations; support for evolving job responsibilities; support for the use of local information; support for divergent administrative practices; support for multiple user groups.

Areas of social knowledge	Social facts
organizational structure	dynamic structure
support structure	variable inter-departmental relationships
administrative activities	evolving job responsibilities
information use	use of local information
administrative practices	divergent administrative practices
user community	multiple user groups

The incorporation of the social facts of the organization into the concept of the administrative processes resulted in a dynamic view of the organization. The dynamic view of the organization was based on social and technical interrelations between employees, information and technology. In the dynamic view, the life of the organization is dependent on change. The Administration software had to be flexible enough to support a dynamic organization. The changes that the software had to support included the following: a) an organizational structure that was dynamic; b) a support structure of the organization that operated on variable inter-departmental relations; c) administrative activities that were based on evolving job responsibilities; d) variable use of administrative information, based on the information needs of separate locations; e) divergent administrative practices, based on the use of administrative information in locations; and f) changing user groups, defined as clusters of work related tasks.

The dynamic view of the organization could continue to flourish by passing the ability to configure the Forms software from the computer scientists to the appointed user representatives. Through conversation with the system designers, the user representatives learned the technical aspects of how to configure the Forms software. By leaving this power, or ability to change, in the hands of the user representatives, the Administration software could continue to evolve with the concept of the administrative processes.

In order to show how the social facts were incorporated into the concept of administrative processes through conversation during the development process, this chapter documents conversations that were held between various project members. These conversations took place between December '90 and July '91. The research data is presented as highlights from conversations that occurred. Relevant organizational texts are also included as research data.

Conditions of this chronicle

Throughout the chronicle, I will follow the use of terms and concepts that referred to the administrative processes to show how they were rendered meaningless as they were taken from the planning phase and applied to the software development environment. Then, through conversation, I will show how the concept of administrative processes were reinterpreted to make sense in the dynamic context of TelCorp.

Summary

The planning phase of the Administration software was conducted in isolation. The systems analysis of the administrative processes was based on formal analyses of the flow of information processes. The systems design of the administrative processes was based on terminology and constructs derived from the GFT software. As such, the terminology and constructs that described the concept of administrative processes was conceived in the abstract.

The configuration of the Forms software involved the reinterpretation of the terminology and construct of the concept of the administrative processes. In configuring the Forms software, the lexicon of GFT-specific terms had to be interpreted in the TelCorp. organization. Also, the assumptions about converting a paper-based administration to an automated administration that were inherent in GFT had to be revised. The terms and software-based construct of the administrative processes had to be reinterpreted to fit an appropriate method of organizing the TelCorp. administration in an automated environment.

Conversation between software designers and user representatives had a strong impact on software development. The impact of conversation was especially evident in the

configuration of the Forms software. When the Administration software was implemented, the concept of administrative processes incorporated areas of social knowledge that had not been accounted for in the planning phase. The result of this revised concept of administrative processes was found in the inscription of social facts of the organization in the Administration software. Once the configuration of the Forms software was complete, the concept of the administrative processes could accommodate flexibility in organizational structure, job responsibilities, inter-departmental relations, administrative practices, use of local information and multiple user groups. The software enabled continued flexibility in the years to come by leaving the configuration of the Forms software in the hands of the user representatives.

Organization of this chapter

The organization of this chapter follows. Background knowledge, Fall 1989 provides a link to the conception of the administrative processes during the planning phase. The origination of the Forms software is based on the Generic Forms Tool (GFT) software. Specifically, this section describes the assumptions about administrative processes implied in the GFT software. Also, terminology is presented that was derived from the GFT software that had to be reinterpreted in the automated environment of TelCorp. for implementation to be successful.

During the Fall 1990, the interactions between software designers and the user representatives were well underway. During this season, the development team was trying to standardize administrative practices. The audience analysis for the Documentation plan indicated that the organizational structure was asymmetrical between geographical regions. The differences in organizational structure implied variance in job responsibilities, use of information, and administrative practices between the regions.

During Winter 1990-91, the configuration of the Forms software was falling behind schedule. It was evident that before the administrative processes could be defined for an automated environment, the user representatives had to establish a way of working together as a group. Furthermore, the user representatives had to establish a way of representing their users. Finally, consensus-building between the user representatives and their users, as well as, amongst the user representatives was established as a review process. During this season the software designers realized that they could not make straight (functional) mapping of the software to operations. This indicated that the functional analysis of users' needs was an inadequate way of defining administrative processes. It was also revealed that the introduction of the Administration software motivated structural changes at the upper levels of the organization. In this case, organizational change was not planned rationally.

During the Spring of 1991, two new user groups were identified. The definition of administrative processes was becoming a reality. There were indications that both the construct and the terminology associated with the social dynamics of the organization were being successfully incorporated into the concept of administrative processes. First, the representation of the network of nodes was gaining acceptance as a suitable structure for the automated administrative processes. Second, GFT-specific terminology was being slowly reinterpreted in the context of an automated TelCorp. However, during the Spring of 1991, user groups were still being defined.

During the Summer of 1991, it was revealed that the tasks associated with the OCB department (control and tracking of the processing of Administration Form 101) were going to be decentralized. The final stage of configuration were included the testing of the distribution path and the definition of the permissions of user groups for the Administration

Form 101. The users' needs continued to be discussed. To prepare for the shift into an automated administrative environment, an implementation strategy had been agreed upon between TelCorp. and R&D Tel.

The Fall of 1991 was the deadline for the delivery of the software to the Field Trial. Right up until the last minute the definition of the users' permission continued to change. This imposed a tight deadline on the production of the Forms software documentation. The "Initiator" user group was created especially to originate the administrative process. The *"Forms Software User Guide"* reflected the revision of the concept of administrative processes. The *"Forms Software User Guide"* also exemplified how the GFT-specific terminology was reinterpreted and revised to suit the automated Administration of TelCorp. The configuration of the Forms software can be modified to reflect any changes in the structure, inter-departmental relations, job responsibilities, use of local information, administrative practices, or user groups in the organization

Background knowledge, Fall 1989

The Generic Forms Tool (GFT) software was developed at R&D Tel. during 1988 and
1989. A document entitled *"GFT Overview"* was written in October of 1989 as a
marketing aide. This was at the same time as the members of R&D Tel. presented this
software at the conference mentioned in Chapter 2 in the section called *"Background
knowledge"*. The following analysis of the *"GFT Overview"* illustrates that the design of
the GFT software contained implicit assumptions about the social dynamics of the
organization that would implemented it. These assumptions were based on a specific
lexicon of terms that reflected the constructs of the GFT software. By describing these
implicit assumptions, I will show how they had to be acknowledged and reinterpreted
through conversation in the configuration of the Forms software.

The implicit assumptions about the social dynamics of an organization that would
implement the GFT software include: software without an intended audience, implicit
assumptions about user groups; implicit assumptions about job responsibilities; implicit
assumptions about administrative practices; implicit assumptions about the use of
information; implicit assumptions about the organizational structure; and, implicit
assumptions about the organizational support structure.

Software without an intended audience

The GFT software had been developed without a customer. Members of the Marketing
division at R&D Tel. were trying to find a paying customer for the GFT software. It was
developed to automate the forms-based administration of any organization. The overview
was used to describe the software to potential buyers. The absence of a target market
underlies the introduction to the document. In a section entitled *"Document audience and
purpose"*, the following is paraphrased:

"This document is provided to give users of GFT software a general overview of the basic parts of GFT. The document is directed at the users of the software. Users should read this overview before proceeding to use the software. If you are a GFT user, read on ...Forms creation and tracking software is new. As a result, you will need to become familiar with a few basic concepts. The purpose of this document is to acquaint you with these ideas. The concepts presented in this document are related to the *design* and *operations* of GFT."[27]

Without a specific audience to write for, the authors of this document adopted a systems development orientation towards the audience. In other words, the potential users of the software were defined in relation to the software features.

Implicit assumptions about user groups

The software is based on pre-conceived notions of groups of people called user groups. The roles of the user groups are implicitly categorized through the software in that these roles are assumed throughout this document. As such, these roles are prerequisites to any organization that would consider purchasing this software. The following quote from the *"GFT Overview"* shows that the software development perspective is based on how the software will support these user groups.

"GFT can be fully integrated to respond to your organization. The system supports user groups by providing selective field protection and visibility, plus user access privileges defined by user group and form type. The person responsible for GFT system configuration pre-defines the organizations which must process a form and the exact sequence of the processes."[28]

---

[27] GFT Overview, pp.1.
[28] Ibid, pp. 3.

89

These user groups are defined in relation to software functions. From the systems development perspective, the roles attributed to prospective users of the software include: a system administrator, a configuration manager, a process user and a monitor user.

A system administrator is a person who is responsible for the day-to-day maintenance of the system. A system administrator is responsible for assigning passwords to new users, assigning access permissions to user groups, making back-ups of the data on a regular basis etc. A configuration manager is responsible for all tasks associated with designing form profiles (designing form templates, designing the form path associated with each template, ascribing read/write permissions to user groups for the fields on each form template). A process user is allowed to access a set of forms and a set of auxiliary software features; also each process user can write information into some of the fields on the forms. The monitor user has access to all forms but would not be able to write information into any of the fields.

Further analysis of the *"GFT Overview"* reveals another assumption about the definition of user groups. A second level description of the users applied primarily to the process and monitor users. The concept of a user community was influenced by the concept of nodes. In Chapter 2 the section called *"Background knowledge"* introduces the concept of nodes while describing the flow of forms in an automated environment.

> "A *path* captures the process for routing form instances. It consists of a set of nodes linked serially or in parallel. Each node corresponds to a particular function which is to be performed. Associated with each node will be a user group responsible for this function. A *group* is a collection of users who typically share a common job function. A user can be a member of more than one group."[29]

[29] Huberman, Tessler, Cameron, <u>A Generic Forms Handling Tool and its Application to Network Trouble Management</u>, pp.3.

A node is likened to a group (of users). A group is a collection of users who share a common job function, yet a user can be a member of more than one group.

The introduction of the concept of a network of nodes initiated a shift in definition of a user community. The roles assigned by the System Administrator, Configuration Manager, Process user and Monitor user were defined by software features on a one-on-one basis. In other words, each category of users could be assigned a set of software functions that they could perform. By enabling a user to be a member of more than one node, the designers were allowing users to perform a diversity of tasks associated with a diversity of software features. Although users were still associated with the software features they could perform, the concept of the network of nodes changed the relationship of the user to the software. Members of the user community could pick and choose between software functions.

Implicit assumptions about job responsibilities

For users to be assigned to a job functions within a network of nodes, job responsibilities must be identifiable and definable. Although users could be members of more that one group, job-related tasks must be defined at each node. Then, job responsibilities of different users would consist of the nodes that they are members of. The introduction of a network of nodes assumed that first, job-related tasks are identifiable and second, that job responsibilities are definable.

Implicit assumptions about the implementation of the GFT software

Implicit in the design of GFT are some assumptions about the how the automated forms software should be implemented in an organization. Because the GFT software is based on a form metaphor, the assumption is that GFT software can be directly implemented into an organization. As such, the paper-based administration would be directly replaced by an automated administration.

In order for a paper-based form to be automated, the following assumptions are made about the implementation of the forms software: a) electronic forms can be designed efficiently from paper forms; b) electronic forms can collect the same information as paper forms; c) forms are used for a specific information-gathering purpose; and d) the path that a form takes through the organization is predictable and consistent.

All of these assumptions must be true for the administrative practices of the organization to remain the same as they were in a paper-based administrative system.

Implicit assumptions about administrative practices

Based on the approach towards implementation, the GFT software assumed that administrative practices would remain the same in an automated environment as they were in the paper-based administration. It follows, then, that if administrative practices remain the same, then so does the use of administrative information in these practices.

Implicit assumptions about the use of information

The use of administrative information is assumed to remain the same through the concept of profile development, or configuration. In the definition and organization of nodes, the administrative processes are defined as information flow between them. Specifically, nodes are identified in relation to how administrative information is partitioned, how each partition is entered; the sequence in which each partition is entered, and when

administrative information is modified. From the system development perspective, these information states are stages in information processing.

Administrative practices are unaltered in the implementation of GFT by assuming that information states are homogeneous throughout the organization. The partitioning of information is expected to be standardized. In other words, once the administrative processes have been identified, they are expected to behave the same in all locations across the organization in all administrative practices.

Implicit assumptions about the organizational structure

Homogeneous information states imply that the organizational structure is symmetrical. If information states are homogeneous, than they are consistent throughout the organizational structure. It follows, then, that functional division, or "families" in this case would be expected to operate on the same information states throughout the organizational structure. The organizational structure is expected to be consistent across geographical boundaries.

Implicit assumptions about the organizational support structure

The view of a consistent organizational structure implied a consistent organizational support structure. In other words, if all divisions are homogeneous in the organizational, then the inter-relations between departments can be expected to be predictable. Departments would inter-relate in a regular manner based on regular patterns of information states. For if administrative practices are predictable then the interaction of departments based on these practices is predictable.

Translation of paper-based administration to an automated administration

Analysis of the *"GFT Overview"* reveals implicit assumptions about the translation of a paper-based administration to an automated administration using GFT. First, because GFT had never been implemented, the implementation strategy was created without the

experience of an actual organization. As such, the implementation strategy was based on traditional practices of computer science which assumed that administration processes could be equated with information processes. Second, it assumed that users can be defined in relation to software (information processing) features. Third, it assumed that the user groups implied by a network of nodes would be readily adaptable to an organization. Fourth, it assumed that job responsibilities are easy to define. Fifth, it assumed that GFT software can be directly implemented into an organization without changing the administrative practices. Sixth, it assumed that information states are homogeneous throughout the organization. Seventh, it assumed that homogeneous information states would operate within a consistent organizational structure. Eighth, it assumed that a consistent organizational structure and administrative practices implied regular patterns of interaction between departments. As a result, the GFT software implied an organizational framework for its direct mapping of a paper-based administration to an automated administration. This organizational framework assumed that the administrative processes could be defined and translated from the existing organizational framework to a construct of a network of nodes without changing existing administrative practices.

Fall 1990

By October 1990, software development was well underway. It was evolving in bits and pieces, at least from my perspective. By now, it was clearly evident that the automation of the administration was a far more complex job than initially visualized following the planning process.

To accommodate software development, an informal project team was assembled. The project team consisted of members from R&D Tel. and TelCorp. The R&D Tel. members of the project team consisted of a manager Stewart D. (referred to as the Development Prime) with seven software designers reporting to him: Joanne P., Jane N., Francis F., Fred S., Dan W., Jennifer L., and Don K. From TelCorp. the project team members included the following: a Tier 3 manager from the Traffic Methods department Jeff B. (referred to as the Project Prime) with a Tier 4 manager reporting to him, Monique R.; a Tier 3 manager from the Transmission Methods department Mike G. with a Tier 4 manager reporting to him, Nadia P.; a Tier 3 manager from the Network Maintenance family, Jonothon T. with a Tier 4 manager reporting to him, Geoff W. Figure 6 illustrates the Administration software project team in the Fall of 1990.

Figure 6

Administration software project team - Fall 1990

Administration software project team

R&D Tel. software designers

	Stewart D. D level manager	
Jennifer L. software designer	Don K. software designer	Joanne P. D level manager
Dan W. software designer	Jane N. software designer	Fred S. software designer
	Francis F. software designer	

TelCorp. user representatives

Jeff B. Tier 3 manager	Andy F. Tier 3 manager	Jonothon T. Tier 3 manager
Monique R. Tier 4 manager	Nadia P. Tier 4 manager	Geoff W. Tier 4 manager

The complexity of the project collaboration was managed with a top-down approach. Stewart D., the manager for the development team supervised the development of individual software modules. Individual software modules were assigned to each developer. Stewart D.'s task was predominantly to oversee the integration of theses software modules.

The first modules developed were software that enabled the building of databases. This was referred to as "conversion" software. Primarily, this software would help the people at TelCorp. to convert their paper-based administration to an automated administration.

During this time, I was designing the end-user documentation suite for the Administration software. For this reason, I was speaking to Stewart D. about how to structure the first end-user documents. There were many conversations with Stewart D. about the design of the documentation suite. The end result of these conversations was the production of a Documentation Plan.

The Documentation Plan for the Administration software specifies the end-user documentation to be written for the upcoming software release. Since I was assigned to this project to write documentation for the Administration software project, the following provides some background to this document which I authored. This section includes background information about writing documentation plans and an audience analysis that I conducted before writing this documentation plan.

Background knowledge

The first stage of technical documentation writing is preparing a documentation plan. Documentation planning occurs at R&D Tel. while software is being developed. When the plan is approved, it is an agreement between the documentation department and the software development department about which documentation will be delivered, when it will be delivered and what resources will be allocated to this project.

The documentation plan specifies the suite of documentation that will be produced in accordance with a release of software. The first challenge in this assignment was to write documentation plans that could make some sense out of the dynamic software development process. Because the Administrative software project extended over several years, and several releases, the planning of documentation was less than straight-forward.

At R&D Tel., technical writers have the difficult challenge of producing "friendly" documentation for scientific staff. Technical writers often start the documentation creation process at the same time as the software is being developed. The developers are their major source of information. Technical writers interview the software developers to get an overall understanding of the purpose and components of the software through open-ended questioning. In fact, a good deal of technical documentation is written from technical specification documentation. The software is usually not available to see until a draft of the documentation is written. At this time, the major structural decisions of the documentation have been made.

Standardizing administrative practices

While I was collecting information for the documentation plan at R&D Tel., the opinion that I heard expressed over and over again, depending on to whom you were talking, was that the administrative processes were being rationalized or standardized. The software designers thought that they were really helping TelCorp. because, through their analysis, they were coming up with techniques to make the administration more "efficient". Efficient, in this context, was understood from the top-down. The following illustrates efficiency from the systems perspective. Stewart D. is the spokesperson for the systems point-of-view.

> Conversation with Stewart D.
> Date: October 24
> S: I think user documentation should mirror the software. Afterall, if the TelCorp. management wants to standardize their procedures then they should follow the software that represents the most efficient way of performing a task.

To produce documentation for Administration software, there were a great many unknown questions that I believed needed to be answered. The information that I had collected from the software designers was oriented towards writing specific documents for each feature of

the Administration software. From their point-of-view, the documentation would be structured based on the functional breakdown of the software features. In contrast, I believed that the documentation should be structured to reflect the use of the software in the context of users' jobs. At this time, there was very little known about how the employees would use the Administration software in their jobs. This was the first indication that there was a real problem with the mapping of the software features into the organization.

To solve this dilemma, I believed that the future users of the software and the documentation would hold the answer. I decided to collect some empirical information about the "end-users" and include the results of this study in a section entitled "Audience analysis" in the upcoming documentation plan.

Audience Analysis

I conducted informal interviews with members of the Training department at TelCorp. who had temporarily been assigned to this project. Their names were Donna F. and Louise C. I held day-long meetings with each of the trainers. At this point, I was interested in who would be affected by the Administration software and what their primary job responsibilities were.

During the meetings I held with the trainers, I learned that the organizational structure of TelCorp. is composed of three main families; Traffic, Transmission and Network Maintenance. Within this context, I was able to find out who was involved in administering the telecommunications network. The results of this information-gathering are paraphrased:

The complexity of organizational mapping can be illustrated through the definition of user groups. The primary user groups are Transmission assignment, Traffic assignment, and Network Maintenance technicians. Each group has special needs.

Transmission assignment is responsible for assigning transmission services and facilities on the telecommunications network. In the Eastern region, transmission assigners are clerks responsible for the design of new circuits on the telecommunications network. In the Western region, the Circuit Designer role is a unique position, and user group. The transmission assignment position could be an entry-level position. However, in each group there is a senior clerk available for guidance and leadership. A small part of this position is currently automated. Expertise is developed on the job.

Traffic assignment is responsible for monitoring the flow of traffic on the telecommunications network. The Traffic assignment clerks ensure that the transmission facilities can accommodate the expected flow of traffic. When overflow occurs on the transmission lines or facilities, Traffic assignment clerks are responsible for re-routing traffic to other facilities. The Traffic assignment group have had little exposure to computer automation. Generally speaking, this group will likely exhibit some fear of automation.

Network Maintenance is responsible for maintaining the physical telecommunications network. Primarily, they connect equipment to the network, and troubleshoot problems effecting telecommunications equipment. All Network Maintenance technicians use computers on the job. This is because they work on the digital switching technology.

Regional differences in job responsibilities and organizational structure

During the audience analysis, I realized that some of the jobs differed between regions. The following is paraphrased:

> The main difference between the regions is in the *structure* of the Transmission family. In the Eastern Region the transmission job functions are specialized, whereas in the Western Region the transmission job functions are generalized. In other words, in the Eastern Region, the transmission job responsibilities are divided between circuit designers and transmission assigners. In the Western Region, these transmission job functions are combined.

> In the Transmission family, the conversion strategy (the strategy for converting paper records to an electronic format) differs between regions. In the Western Region, paper records will be converted during light work loads. In the Eastern Region, contract people will likely be hired specifically for conversion. The conversion strategy is also discussed in the Audience support section of this document.

To summarize, the audience analysis that was conducted for the Documentation Plan concluded that there were three separate user groups of the Administration software and that one of these user groups had two distinct user groups within it.[30] User groups were defined by the major job responsibilities that the users would engage in throughout the day. Within the Transmission family, jobs were split up in different ways. The reporting structure within each region reflected the job responsibilities. The differences between the sub groups within the Transmission family could be attributed to structural differences between two organizational regions.

---

[30] It should be noted that conversations with the developers, at this time, did not include any specific needs of end-users. To the developers, any software users are just that: software users. The context in which people use the software are not usually taken into account.

Variable administrative processes

When I brought up these structural differences within the Transmission family, via the distribution of the Documentation Plan, I was met with annoyance. These differences between users' needs were an irritation to the developers' view of how administration should work. Until this issue was written down in the Documentation Plan, these differences had been categorized as irregularities. As irregularities, they weren't expected to cause any major reconceptualizing of the administrative processes. This was another indication that the software designers viewed administrative processes as consistent and homogeneous.

A structural difference in the organization is hardly an irregularity. Once it was written down on paper, it had to be fully addressed in the design of the documentation, and in the design of the software. These structural differences implied that the administrative processes within the Transmission family would vary between regions. If job functions were generalized in one region, yet specialized in another region, then the administrative processes would vary between regions. As such, administrative information would be used in different ways between regions.

Use of information in administrative processes

The use of information can vary between locations for either of two reasons. First, as previously stated, if administrative processes are different between locations then information will be used to accommodate the administrative processes. In the case of the Transmission family, for instance, if job functions are specialized in one region, the administrative information will be used for many specific tasks by several people. Where job functions are generalized, the administrative information will be used by one person doing several tasks. The use of information can also vary between locations if the same information is used for different reasons in different locations because it has a different

significance. For the purposes of this essay, this is the case of 'local meanings" where information is determined by situation-specific contexts. In a note to myself dated December, 1990, the following is paraphrased:

> In a conversation I had with Donna F., a Corporate trainer, she revealed that local meanings create havoc in her preparations for training courses. In the case of automation where practices are standardized across regions (the corporation divided into geographical areas) local meanings are associated with everything from names to occupational practices and processes.

> Trainers are also concerned with how people situate their jobs. The new methodology of organizational training, they say, believes that tasks are performed based on the information requirements of the local organization. Through an analysis of information needs, Donna has found that not only are there problems associated with local meanings but also, divergent information needs from the same organizational processes. Different local organizational contexts can have different information needs for the so-called "same" organizational procedure.

Since the corporate training budget was slashed, Donna F.'s knowledge of these kinds of problems was never shared with the software designers. Instead, alterations were considered to be problems of deviations from a standard. I would always remember this conversation, as variance in information needs of the users would appear often.

Seasonal summary

Through conversation, the Fall of 1990 revealed the following developments in the conceptualization of administration processes:

• Job responsibilities vary between regions.

• Administrative activities vary between regions.

• The organizational structure varies between regions.

• The use of administrative information varies between locations.

Winter 1991

By the end of 1990 there were danger signs about the late start of the configuration of the Forms software. This is indicated on a Monthly Progress Report. The report highlights the current month's progress achievements on the project, the plan for next month, and highlights any concerns associated with the project. In December, 1990, the Monthly Progress Report for the Administration software highlighted the following under the heading "Concern":

> The configuration of the Administrative Form (101) is on the critical path for the deployment of Administration software 1.1.

At this time the activity of configuring the Forms software was supposed to have begun. The concern was that it had not been started. Actually, the configuration of the Forms software was not being addressed by anyone on the Administration software project team. No one at TelCorp. was taking any responsibility for the major tasks identified with the configuration. These tasks included the following: design of the template for the Administration Form (101); creation of the user groups or nodes; and the design of the distribution path for the Administration Form (101).

By highlighting that the "configuration" was on the critical path, the management of R&D Tel. (Stewart D. and his boss Steve M.) was telling the management of the TelCorp. that unless this activity was initiated, the implementation of the Administration software would be jeopardized.

Up until this time, software designers on the Administration software project team from R&D Tel. had been working on the development of the Forms software in isolation. The user representatives on the Administration software team from TelCorp. were not involved

in the design of the forms to be accessed from the Forms software. Although, it was TelCorp.'s responsibility to configure the Forms software, they were late in starting.

The late start indicated that it was not just the technical aspects of the configuration that had to be addressed. As the months rolled by, it was evident that before the user representatives could configure the Forms software, they would have to agree on what the administrative processes were. As the following sections will illustrate, agreeing on the definition of the administrative processes entailed a great deal of collaboration between people who had no history of working together. The interactions between the Administration software project members was a strong influence on defining what the administrative processes were at TelCorp., and how the administrative processes would work in an automated environment.

Effect of department and inter-departmental relations on defining administrative processes
By the end of January, the frequency of interactions between R&D Tel. and TelCorp. was increasing rapidly. Sometimes people met on a one-to-one basis. More often, the entire Administration software team met. It was on these occasions that I noticed that the inter-departmental relations between the user representatives at TelCorp. were not smooth.

The relationships within the same methods departments weren't exactly smooth either. For instance, the relationship between Jeff B. and Monique R. was an interesting one. Jeff B. usually represented the users to the upper management of TelCorp. He would defend the users' interests in selling the idea of the Administration software to his management in terms of how it would benefit the users. Monique R., on the other hand, actually went out to the users in the three families to talk with them. There was a great deal of "leg work" involved in working with the line groups to seek their approval on the software. At each step of the development process, Monique R. had to gain the written approval of various

line group committees. She also had to act as a mediator between R&D Tel. and the line groups. Through all of this negotiating, Monique R. had assumed a subservient role in her relationship with Jeff B. The following are notes taken during a meeting for the Administrative software on January 28:

> Jeff B. says "We need to define detailed activities because a lot of people are waiting for this project." and "We've got to work together as a team." After this enthusiastic introduction, Jeff B.asks Monique R. to take notes. Monique R.is sitting beside me. She rolls her eyes and takes out a pad.

Throughout this meeting, Jeff B. pleaded for the members to act like a team several times. This indicates the seriousness of the relationships between the project players. There are undercurrents of strain between all players of the Administration software project team.

Control and coordination of administrative information

A part of the strain can be attributed to new tasks arising. Nobody wanted to take responsibility for more work. The following journal entry is dated January 28th:

> The Forms software is designed to enter records into one of the Administration software databases. Before the Forms software can enter these records automatically, the current records have to be entered into this database. A specific tool is being developed to enter the current records. The conversation today was mostly a dispute about who the tool should be designed for and, hence, who was going to perform the eventual task of entering the data. The consensus seemed to be that the Transmission clerk is familiar with the information or data contained in the record. They are familiar with the History of the forms. But the full information isn't on the form until it reaches the Traffic department. Nadia P. (Transmission dept.) believed the Traffic clerk should enter the records.

Coupling of Traffic, Transmission and Network Maintenance families

The interactions between the Traffic and Transmission families usually happened in pre-defined ways. The areas of their responsibilities were distinct. The introduction of new technology, however, brought people from these families together in a way that was new. This was the first time members from the three families had worked on a project together.

There were two main reasons for coming together. First, for the administration to run smoothly, they had to establish how their families would use the Administration software once it was implemented. Second, between them, the user representatives had to clarify the administrative processes that were to be automated in the software. With regard to the latter, issues arose pertaining to the rules governing the issuance of the Administration Form 101, when these forms could be sent from one department to another, under what conditions these forms could be modified, when these forms could be distributed, etc. Every single aspect of how the Administration Form 101 was routed around the organization had to be agreed upon among the three user representatives These forms actually drove the administration of the telecommunications network. The Administration Form 101 kept track of any changes to the network, and any change implied some type of activity in each of the families. There were many situations that had to be discussed.

For instance, the three user representatives had to agree on when Administration Form 101 could be issued. Again from the January 28th journal entry:

> The issue was brought up about when an Administration Form 101 could be issued. Under normal circumstances the Traffic department would issue the Administration Form 101. However, a situation was raised where the Transmission department would issue the form. The issue seemed to represent the relationship between the Traffic and Transmission departments Whose responsibility is it to instigate the Administrative Form 101?

PAGINATION ERROR.   ERREUR DE PAGINATION.

TEXT COMPLETE.   LE TEXTE EST COMPLET.

This scheduling had caused some resentment. The resentment was evident in their infrequent attendance at meetings and in their attitudes at these meetings. In one such instance, both representatives from the Network Maintenance family were there. This was unusual, since Jonothon T.usually sent Geoff W. on his own. They were involved in a test of another piece of software. They had run into some technical problems so the test had stopped temporarily. Jonothon T. expressed frustration with a threat in the following journal entry:

> Jonothon T., always a lively man at meetings, says "I'll raise it to the 3rd level (a higher level of management) or give me the cash to resolve it."

The new relations between the user representatives were not always discussed rationally.

Consensus-making between the Traffic, Transmission and Network Maintenance families

The ability to arrive at a consensus between the Traffic, Transmission and Network Maintenance families was absolutely vital to the configuration of the Forms software. The definition of, and inter-relations between, User Groups entailed a great deal of cooperation and collaboration between the user representatives of the three families.

The definition of User Groups entailed four kinds of agreements:

a) since all users (over a thousand) had to be associated with groups, the families had to agree to the *conditions of group formation*; b) since the definition of User Groups had to mirror the internal structure of existing groups, the families had to agree upon *when the groups would interact*; c) *privileges* had to be assigned to each group determining which forms each group could access, who could make changes to forms, and which other Administration software tools users could access through the forms software; and d) since each form followed a distribution path through the network of nodes, the *sequence of this distribution path* had to be agreed upon between the user representatives.

As a group, the user-representatives of the three families had really not established a way of arriving at a consensus. The decisions involved in how the administrative processes were to be automated touched on all aspects of routing the Administration Form 101. Since this form involved clerks or technicians from the Traffic, Transmission and Network Maintenance families, they had to agree on the methods implied in the configuration of the Forms software. Otherwise, it was not guaranteed that the Administration software would work as an effective administration tool. As user representatives, this was part of their job. The difficulties that they faced in arriving at a consensus were many.

One type of difficulty in their approach to making these decisions is illustrated in the construction of the distribution path. The construction of the distribution path implied: i) who sends the form to whom, and ii) what will the receiver do with it when s/he receives it? To illustrate the difficulties surrounding the construction of the distribution path as of January 28th, my journal states:

> The rules governing the distribution of work orders need to be articulated and coded into the configuration of the Forms software. The issue arises as to who will make these decisions. If Transmission decides then Network Maintenance has a dependency on them. This is not a position they want to have. They are a hard-tech bunch of guys.

In order to arrive at a consensus about what the "rules" should be, each of the user representatives need input from the users of all three families. This was the first time all three user representatives required input from their families. Until now, they had gone to their users for specific questions on their own volition. From the way they responded, it seemed to be a different matter to be required to come back and report about a consensus among their users.

The prospect of deriving a consensus between the three families about these rules seems highly unlikely. At the beginning of January, the three families do not seem to have the experience or the inspiration to discuss the business processes together. This is a real problem. My journal states:

> Everyone feels the pressure of the past 2 years and next two years building on their head. Also, there is a lot of responsibility that nobody seems to want to take. The inter-departmental team approach is not supported on any other level. Each of the TelCorp. participants seems to relate vertically. They can look up and they can look down but looking horizontally seems to put a crick in their necks.

In the end, Monique R. is assigned the responsibility of configuring the form. She will design the layout of the Administration Form 101 and list the user groups for the distribution list. When it is fully configured, she will send it to the user representatives of the other families to collect their feedback. They can use the draft of the Administration Form 101 as a method of collecting their feedback. **Consensus-building is, therefore, established as a review process.**

Forms configuration initiates promotion at R&D Tel..

In February Joanne P. was promoted. She became the D-level manager in charge of the development of the Forms software module. My journal notes the following:

> Joanne P. was promoted. I heard this second-hand this morning. Later in the afternoon, I went downstairs to congratulate her. She said that the rationale for creating a new position was to "get the clients to take the configuration of forms more seriously."

Until this time, the Forms software was just another software module under the responsibility of Stewart D., the software development manager for the Administrative software team. Now, Joanne P. and Stewart D. were at the same managerial level.

Figure 7 illustrates the structure of the Administration software project team after the promotion of Joanne P.

Figure 7

Administration software project team - Winter, 1991

Administration software project team

R&D Tel. software designers Administration software group	R&D Tel. software designers Forms software group	TelCorp. user representatives

Stewart D D level manager		Joanne P. D level manager	Jeff B. Tier 3 manager	Andy F. Tier 3 manager	Jonothon T. Tier 3 manager
Jennifer L. software designer	Don K. software designer	Fred S. software designer	Monique R. Tier 4 manager	Nadia P. Tier 4 manager	Geoff W. Tier 4 manager
Jane N. software designer	Dan W. software designer				
Francis F. software designer					

R&D Tel.'s reflections on the inter-relationships between the Administration software project team

Joanne P.'s promotion clearly indicated to me that the Administration software project team was in a state of panic. I spoke with Joanne P. soon after she was promoted In the following, Joanne P. reflects on the original project planning:

> J.P.: We made a mistake in the beginning. R&D Tel. grossly underestimated the amount of time and resources the project would take. And from there, that view was never altered.

> *M: What is the view of automating, everyone seems to think that it will be easy?*

> J.P.: Well this is new. We've always had political problems on this project but now the people who are supposed to look after implementation and training are fed up. They are sarcastic about the proposed schedules. This is new. This is what makes me nervous.

With respect to the configuration of the Forms software, there was still a great deal of work to be done. Monique R. and Nadia P. would be working together with Joanne P. until the configuration was complete. The planning of the Administration software project had been way off. If the project was to succeed, it was evident that the team had to cooperate. Geoff W. was rarely showing up for the Administration software project meetings. The responsibility for configuring Forms software was left with Monique R. and Nadia P. I wondered what Joanne P.'s view was of the relationship between Monique R. and Nadia P.

> *M: How do Monique R. and Nadia P. get along?*

> J.P.: Well in the beginning it was pretty good. They were cooperative. Sometimes they've gotten to the point where they couldn't even talk to each other. I think its a matter of pressure. They need to work together on the Forms software. Nadia P. told Monique R. it was her job. Monique R.

from the beginning assumed the responsibility for the Form software. Nadia P. won't take ownership of it.

Reflections on users needs

At the beginning of April I conducted informal, interviews with both Jeff B. and Joanne P. I asked both of them about the need for the Forms software. At this time, the definition of administrative processes through the configuration of the Forms software seemed like such an ordeal that I wondered if there really was a need for the configuration of the Forms software, or for that matter the Administration software itself.

> *M: Is the Forms software fulfilling a need?*

> J.B.:   Yes, it is saving time and eliminating redundancy.

> J.P.:   (Yes, but) It is touching on their working processes. We need their agreement (to define the configuration).

I wasn't sure what the need was. So, I took a different approach. Maybe the eventual software users really needed this software.

> *M: Is this need perceived by the people who will use the software?*

> J.B.:   Yes, it is obvious.

> J.P.:   Their mode of operations is complicated. We can't just make a straight mapping of the software to operations. A lot of the work on paper is too hard to map.

Maybe there are different types of needs.

> *M: How do you decide what needs should be met?*

> J.P.:   You can only get a bit (of what the needs are) at a time.

J.B.:   Each department has their own priorities based on the technology
they have available to them. Each departments has their own view of their
users' needs. Every department will hang on to their own priorities to
protect themselves.

Traffic's priority is the inventory of trunk terminations. Transmission's
priority is the automated data capture from the existing transmission system.
Currently, clerks have to transcribe this information. Network
Maintenance's priority is to access to electronic inventory contained in all
the databases connected to the Administration software.

Their responses told a different story than the *"User Needs Requirements"* document that
was produced during the planning phase. (See *Chapter 2: analytic foundation* for an
analysis of the *"User Needs Requirements"* document.) Joanne P., representing the
software development company, expected users' needs to come packaged as a set (like the
"User Needs Requirements" document). On the other hand, Jeff B. looked at each of the
families as a group of users, each having their own definable needs. At this point in the
software development process, Jeff B.'s attitude was shifting about users needs.

User needs were perceived in two different ways. The software designer (Joanne P.)
perceived users' needs as a set of functional requirements from a systems perspective. The
user representative (Jeff B.) perceived users' needs as they related to individual, and
sometimes competing departments.

The users' needs propelled the definition of the administrative processes. Unfortunately,
this was not how they were perceived. The (functional) framework of users' needs pitted
the families against each other. (For a complete analysis of the functional analyses of
users' needs see the section called *"User Needs Requirements"* in *Chapter 2: analytic
foundation.)* It was no wonder that no one at TelCorp. had a global view of administrative

processes. In taking this approach to the definition of administrative processes, the families were set up to war with each other.

Reflections on the relationship between the planning and implementation of technology

My next question related to the management of change in the customer's organization. I reasoned that the difficulties of consensus-making between the user representatives were great. The effect of the implementation of the Administration software on the lower levels of management, up until this point, seemed to be that it was causing a lot of grief. If the implementation of new technology was so difficult for the lower levels of the customer's organization, then what was the policy for managing technological change in the organization?

> *M: Are there any policies relating to implementing software in TelCorp.?*

> J.B.:   No. Change is not happening through rational planning.

This response surprised me, after all the planning documents that were created to get this project off the ground. Jeff B.continues his explanation:

> J.B.:   Initially a study was conducted called the Network Operations Administration Management Information System Planning. This study identified that each clerk is maintaining personal logs about the work orders that pass by their desks. They kept these logs to account for their time. The study said that these were redundant databases (of records). The Administrative software is in line with their recommendations. I used this document as a basis for the rationalization for the Administrative software.

So change had been initiated through a study. The recommendations from this study were to automate the administration. But that project had never gotten started. Jeff B. didn't have an explanation for this.

Preparing for structural change through the Administration software

The difference between the system proposed in the *"Network Operations Administration Management Information System Planning"* study and the Administration software project, was that the latter was preparing for organizational change in the future. Jeff B. believed that this was the reason that the Administration software project had gained acceptance from the upper management at TelCorp. Jeff B. reflected:

> There is resistance to change in the organization. Especially at the lower levels. However, the Administrative software is enabling organizational change in its design.

The Administration software was preparing for change in its design. This was evident through the actions of upper management. Jeff B. continued:

> The Administrative software has been publicized through various communication channels throughout the corporation. (Based on these communications) some AVP's are preparing their departments for further change. Organizational change is happening to prepare for the implementation of the Administrative software. For instance, at the C and B levels, the lines between Transmission and Traffic departments is (already) dissolving.

The implementation of technology is causing structural changes to the organization, without a coherent policy to guide it.

Organizational change at the upper levels

Structural change is occurring at the upper levels first. The people at the lower levels don't even know that there is a relationship between the implementation of the Administration software and structural changes in the organization. Until the structure has changed at the lower levels, the clerks and technicians won't know that a change in the structure of the organization is happening. At this point, there is no indication for them to know that a change in the organizational structure is occurring.

It was still not clear to me what the relationship was between the Administration software and structural change in the organization. Moreover, how could the Administration software enable a structural change of this magnitude? I could see how the idea, or concept, of automated administrative processes could motivate the upper management to act. For they would have been involved in the approval of the funding for the project to begin with. Funding for a project of this kind is managed at the AVP level. Yet, how could the day-to-day operations of the telecommunications giant be restructured? Jeff B. clarified this question. From my journal, I've noted the following:

> Jeff B.'s rationalization (for how the Administration software is enabling change in the organizational structure) is that because the Administrative software is divided into discrete functions, job descriptions can be easily changed. In the future, perhaps three to four years, the responsibilities of the Traffic and Transmission departments will merge.

Seasonal summary

Through conversation, the Winter of 1991 revealed the following developments in the conceptualization of administration processes:

• Before the administrative processes could be defined for an automated environment, the user representatives had to establish a way of working together as a group. Also, the user representatives had to establish a way of representing their users.

• Consensus-building between the user representatives and their users, as well as, amongst the user representatives was established as a review process.

• The software designers realized that they could not make straight (functional) mapping of the software to operations. This indicated that the functional analysis of users needs was an inadequate way of defining administrative processes.

• The introduction of the Administration software motivated structural changes at the upper levels of the organization.

• Organizational change was not planned rationally.

118

Spring 1991

By May, the configuration of the Forms software was taking on a real sense of urgency. The most recent schedule that I had heard was that the training for the Forms software would take place in August. At the end of the month of May I happened to run into Monique R. at a project meeting. The following is from my journal of May 24:

> I took the opportunity to talk about the progress of the Forms software with Monique R. She was quite excited about the progress of the Administration Form 101 configuration. I told her that I would need her support through the month of June. She told me that she was preparing some technical specifications (of the electronic forms) for me that would be useful. She also said that she thought Jeff B.'s rollout schedule was idealistic. She imagined that the invitation for training would be sent out and no one would reply because of vacations.

An unanticipated user group

During May, R&D Tel. was still studying existing paper forms to see how they should be implemented in the Administrative software. As they were analyzing the paper forms, they were trying to think of ways to rationalize the administrative processes. In an effort to standardize the administrative processes during May, Stewart D. suggested that perhaps the Transmission group should take full responsibility for issuing the Administration Form 101. As my journal illustrates in the following, this suggestion brought out a new user group, Traffic Provisioning:

> Stewart D. was trying to suggest the Transmission take on the responsibility of making the assignments, without the notification of Traffic. Nobody at TelCorp. liked this idea. The reason is that the Traffic assignment people are in a position to make the decisions based on the information that they get from the Traffic Provisioning people. Provisioning draws up a map, or a plan, in the new system all that work and forecasting is going for nought.

Until this time, the function of Traffic Provisioning had not been discussed. Neither had they been designed into the Administrative software. This was a new user group.

A second new user group

Three days later, I was invited to a meeting to discuss the creation of the form paths. Aside from Monique R., Joanne P., Stewart D. and Fred S., there were two new faces in the room (Nancy V. and Georgia H.). I soon found out that these two women were taking over Nadia P.'s job. Nadia P. was taking an early retirement.

There was a sketch of a distribution path for the Administration Form 101 on the blackboard. The path that was discussed is illustrated in Figure 8.

Figure 8
Distribution path
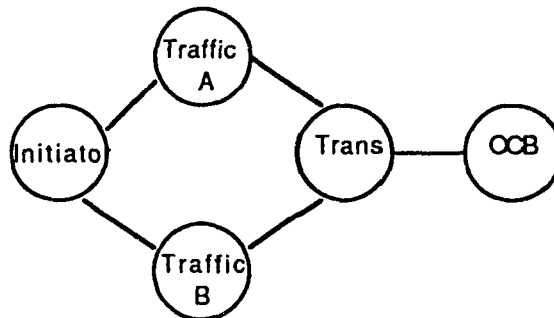
I was shocked that OCB represented one of the nodes. Who were they? This was supposed to be the last node in the path. But nobody really knew what they did. Their official name was the Order Control Bureau, however everyone referred to them as the Control department. I wasn't even clear on which family they belonged to. It was agreed that a meeting would be set up in June so that the Control department could specify their needs. This struck me as rather late in the process.

Although another new user group had just sprung up, the discussion of the path of nodes was a pleasant surprise to me . This was the first time that the GFT-specific terminology had been used to describe the administrative processes of TelCorp.

Everyone seemed to pick up on the representation of departments and user groups as nodes with ease. To me, this was an indication that the node symbols worked well to describe groups of different kinds. My journal includes the following entry:

> The nodes represent the Initiator (either Traffic or Transmission), Traffic assignment, Transmission assignment, and OCB . However, the groups within these nodes are still not identified. At one point, it was discussed that different people in Transmission often needed to pass an Administration Form 101 back and forth. Joanne P. maintained that this was possible. There is a function called, "Refer" which allows the user to refer a copy of a form to someone in another group.

The job, at this point, was to agree on the paths of distribution for the Administrative Form 101. To do this, the activities of each node had to be specified. Each activity would be performed by one of the other software tools include in the Administration software. (See *Chapter 2: analytic foundation*, the section called *"Project Management Plan"* for a complete list of all the software tools contained in the Administration software.) Each node would be attributed specific access permissions to the other software tools. The access permissions identified at what point on the "path" certain administrative functions could be performed.

Monique R. had drawn ten different paths to accommodate ten types of administrative processes that had been identified in the analysis of the forms. Each path included a definition of which nodes the Administration Form 101 would be on the distribution list, as well as, the activities that were performed at each node.

Joanne P. pointed out that these activities were usually associated with existing departments at TelCorp. That is why most nodes reflect the name of a department. However, each node would be associated with user groups. The user groups were, generally speaking, sub-categories of the departments. For instance, the Traffic provisioning department was a sub-category of the Traffic node. Specific user groups, based on job-related tasks, still had to be defined.

Seasonal summary

Through conversation, the Spring of 1991 revealed the following developments in the conceptualization of administration processes:

• Two new user groups were identified.

• The representation of the network of nodes gained acceptance as a structure of the automated administrative processes.

• GFT-specific terminology was being successfully reinterpreted in the context of an automated TelCorp.

• User groups were still being defined.

Summer 1991

Establishing the role of OCB

Here it was, the end of spring, and the user groups had not been defined. Nobody had an idea of what activities the OCB department was involved in. It amazed me that in the definition of the administrative processes, the role of an entire department had not been addressed. It was evident that during the systems analysis phase, none of the software designers nor the user representatives had ever asked, what does the OCB department do? Defining the configuration for the Forms software would certainly have been simplified if it had.

At the end of June, I was invited to a meeting to address the role of the OCB department . My journal notes the following:

> In her introductory remarks, Monique R. referred to the OCB as "the policeman of management reports".

It soon became clear that the OCB department had a dual role. The first was to track the Administrative Form 101 through the processing cycle. The second was to ensure that the processing of each form conformed to predefined rules about how long each stage of processing could take.

The question arose as to who takes the control responsibility for each form, or how is the responsibility for the control of forms decided upon. The following is from my journal:

> A funny statement came out of a conversation about Form control. "Control goes to the A office. If there are 2 A offices, then control goes to the 1st in alphabetical order. I know how stupid that sounds, but that's the way we do it."

Multiple uses of local information

There were members of the Traffic, Transmission and OCB at the meeting at the end of June (Network Maintenance was conspicuously missing). It didn't take long to see that OCB used the information of the Administration Form 101 for very different uses than had previously been expressed by any of the other Administration software project members. For instance, dates on the Administrative forms are used by OCB to track the forms. From their perspective, the dates should only be made available to them. Members from both the Traffic and Transmission departments had trouble with that idea. My journal includes the following entry:

> Transmission asks for the system to generate dates to tell the assigners how long they have to work on an order.

Dates are used by each department to manage the flow of local administrative work. This was a great example of how information has different functions and multiple meaning in different contexts. The dates on the Administration Form 101 were clearly situation-specific. Each group used the information on the form for their own use. Not only was the Administration Form 101 used to administer the telecommunications organization, but it was also used as an internal regulator of the productivity of each department involved in administration.

The changing role of OCB

During this meeting at the end of June, there were utterances about the changing role of OCB. Someone even mentioned that they would soon be eliminated. The following journal entry captured a conversation with one of the OCB members from the Western Region:

M: *Are you really trying to eliminate your (OCB) position?*

G: Yes

M: *What is your interest in doing that?*

G: I'm trying to decentralize control functions into the Networking Maintenance group. There is a part of Networking Maintenance that does some admin functions, so they can carry these out too.

M: *How did you find yourself in this position?*

G: I always do this.

M: *Excuse me?*

G: I'm good at it. I get sent into a department, reorganize, standardize, centralize and (in doing so) try to eliminate some positions.

M: *Where do the people go?*

G: People never lose their jobs at TelCorp. We send them to areas that are growing.

The changing organizational structure of TelCorp.

From the conversation noted above, it appeared that some effort was being put into preparing for the reorganization of the OCB department. This effort, though, had not been known by either the software designers of the Administrative software, nor by the user representatives of the Administration software project team. After all, this was the first time the OCB department (including the OCB Methods people) had been contacted about the implementation Administrative software. Coordination amongst the Administration software project team had truly broken down in the planning phase.

The case of OCB also indicates that changes in organizational structure are happening at TelCorp. above and beyond changes effected by the implementation of technology. This indicates that the changes to the structure at TelCorp., are not caused by the introduction of the Administration software. Rather, new technology should accommodate changes to organizational structure. In this case, the move towards the decentralization of the OCB-activities could certainly be accommodated by the Administration software. The technology is not limited by the organizational structure, but the analysis and the design of the administrative processes did not even acknowledge that changes to the organizational structure were being discussed during the summer of 1991.

Finalizing the Forms software configuration

During July and August, there was little interaction between the members of R&D Tel. and TelCorp., although everyone, for their own part, was working towards implementing the Forms software. At the end of August, the forms were still being configured. The distribution paths were being tested. New paths were tried out. In fact, Joanne P., collapsed the ten paths that Monique R. had drawn into one. The layout of the Administration Form 101 was being constantly revised. Each time Monique R. revised the forms she would send it to Joanne P. to test.

Mid-September was the deadline to start the Field Trial. The official software development process suggested that, by the end of August, the forms should be configured so that they could be tested for two weeks before going out to the Field Trial. The Forms software was unusual in that there was no date to go to testing. As a fully developed software product, GFT had been through the entire software development process. The official software development process included testing. Since GFT had been tested, the software developers reasoned, there was no need to have the Forms software tested.

Without this internal deadline (in the software development process) the software seemed to change on a weekly basis. In my capacity as a technical writer trying to document the software, this made life very difficult. It was like trying to draw a moving picture-very difficult to capture without standing still.

At the end of August the Forms software was in a fragile state. For one thing, the software wasn't always accessible. I was called to a meeting at TelCorp. to review the latest version of Forms software. Since the Administration software project members had all reviewed the first draft of the Forms software documentation, we were scheduled to discuss this review.

> We started the meeting at 10 am. The location was at TelCorp. We were supposed to look at the most recent version of the Forms software in the morning and then the documentation in the afternoon. Surprise, the "load" or version of the software had a glitch. So we couldn't look at the Forms software.

Flexibility of administrative processes

The fragility of the software could also be attributed to fact that the permissions of the user groups were still being defined. At this time, there were still discrepancies between the access permissions and what each user group really needed in their working practices. My journal notes the following:

> The Transmission user group isn't supposed to have access to the trunk service. However, sometimes they have to change the trunk identification. This feature is only accessible through the trunk service. This is another case of how users' needs are not anticipated very well. This is rather late in the game to find out they need to be able to modify this information. This information, until now, has been contextualized as Traffic information.

It should be noted, that the ability to alter the permissions of user groups was, in the end, a blessing in disguise. Although there had been many difficulties discussing what these permissions should be, it was a real benefit to be able to modify these permissions.

Last minute definition of users' needs creates uncertainty

Finally, the fragility of the Forms software could be attributed to the fact that new user needs were still being talked about. As the deadline for bringing the software out of the laboratory approached, new user needs were creating great uncertainties. My journal notes:

> Also, the story of who uses what for what seems to go on forever. We keep learning that people and systems have different kinds of requirements for information. Once Jeff B. and Monique R. saw how many fields the initiator has to fill in on the Administration Form 101, they finally wanted to cut some of them away. They seemed paralyzed by these demands for information from different contexts.

Another example of the tension surrounding users needs follows:

> Nancy V. has made a request about one of her users. Monique R. reports that Jeff B. has told her not to start taking "special" requirements from "the user". After all, there are too many users to worry about. User requirements are about groups, not individual people. Finally, Monique R. says with disdain that "There are no savings to Transmission with the Administrative software." Both Monique R. and Nancy V. roll their heads. Apparently, the last economic study conducted by the Transmission people shows that there are no savings with the automated system. They have reorganized the division since they found out that the automated system was on its way. Consequently, Monique R. is put in a rather awkward situation. She says "We've made lots of changes to the system specifically for them...they've blackmailed us."

The users' needs had changed radically since they had been defined in the "User Needs Requirement" document. (See Chapter 2: analytic foundation for a complete analysis of the "User Needs Requirements" document.)

Implementation strategy for the Administration software

The implementation strategy that was agreed on by all members of the Administration software project team, was called parallel processing. This meant that both the electronic and the paper Administrative systems could be practiced simultaneously. There were two main reasons for this. First, the Network Maintenance family had been taken out of the first implementation scheduled for 1992. It was too expensive to deliver the terminals to all of their locations. As such, the Traffic assigners and Transmission assigners would print out the electronic forms and send them to the Network Maintenance technicians. Network Maintenance would work from these hard copies. When they had completed their work, they would send the hard copies to the person responsible for the Control function. The Control person would enter the completion date for the Administration Form 101 into Administration system. The second reason for using a parallel processing implementation strategy was that, in the end, the routing path could not accommodate all types of administrative circumstances. My journal notes:

> The issue of parallel processing came up. It seems that the Forms software will not be able to handle all types of Administrative forms. For all the analysis that the developers did, they couldn't come up with an automated system that could accommodate all possible types of forms, or work orders. So, they are saying that the organization will accommodate parallel processing. What they mean is that both an automated and a paper-based administrative processing will occur simultaneously.

Seasonal summary

Through conversation, the Summer of 1991 revealed the following developments in the conceptualization of administration processes:

• The tasks associated with the OCB department (control and tracking of the processing of Administration Form 101) was going to be decentralized.

• The distribution path for the Administration Form 101 was being tested.

• The Forms software was in a fragile state.

• The permissions of the user groups were still being defined.

• The users needs had changed radically since they had been defined in the *"User Needs Requirement"* document.

• An implementation strategy had been agreed upon between TelCorp. and R&D Tel.

Fall 1991

The head of the R&D Tel, laboratory (Alexander H.) called us all together. He announced

that Stewart D. was promoted to "C" level. For now, there is no replacement for

Stewart D.'s D level position. Figure 9 illustrates the Administration software project team

after Stewart D.'s promotion.

Figure 9

Administration software project team - Fall 1991

Administration software project team

R&D Tel. software designers Network Operations group		TelCorp. user representatives		
Stewart D. C level manager				
Stewart D. D level manager	Joanne P. D level manager	Jeff B. Tier 3 manager	Andy F. Tier 3 manager	Jonothon T. Tier 3 manager
Jennifer L. software designer	Don K. software designer	Fred S. software designer	Monique R. Tier 4 manager	Nadia P. Tier 4 manager	Geoff W. Tier 4 manager
Jane N. software designer	Dan W. software designer				
Francis F. software designer					

131

During the Fall of 1991, the focus of the Administration software project team was directed towards the Field Trial. Aside from finalizing the configuration of the Forms software, a user guide had to be produced for the Field Trial. As the author of this documentation, I was privy to background knowledge about its production. The following provides some additional information about the production of the document entitled *"Forms Software User Guide"*.

Background on the production of the "Forms Software User Guide"
The deadline for the production of the Forms software documentation was impossible. The second draft of the document (the Preliminary issue) had to coincide with the beginning of the Field Trial. The document had to be available for the members of TelCorp. for the software training phase of the Field Trial which was scheduled for the beginning of the Field Trial. The Field Trial date had slipped before this, but September 21st was the last call.

Neither the Forms software nor the end-user documentation that went along with it was being developed for all the anticipated Administrative Forms of TelCorp. Of the two targeted Forms (Administrative Form 101 and Administrative Form 102) only the processes of Administrative Form 101 were automated with the Forms software. The tasks associated with configuring a single Administrative form, although certainly a primary administrative form, were far greater, more time-consuming than had been anticipated by either R&D Tel. or TelCorp. At this point, the plan was to test out the automated Administrative Form 101 with a series of possible cases in the Field Trial, modify it with the comments collected in the Field Trial and then proceed to automate Administrative Form 102. The *"Forms Software User Guide"*, provides illustrated procedures for the operations of Forms software as it has been configured for Administrative Form 101.

Consequences of user-centred design

The production of this document was more pressure-driven than usual. The main reason for this was that the definition of the users' permission changed radically by the user representatives up to the very last minute. This is work that software developers usually do, from the perspective of a technical writer. The software developers are at least oriented to finalize changes in the textual elements of the software interface at least two weeks before an end-user document is produced. This gives the writer enough leeway to complete all the work involved in publication. In this situation, because the Customers (TelCorp.) were responsible for the textual interface, they were unaware of the needs of other groups in R&D Tel. (like the technical writers), and could not deal with anyone else's constraints or demands.

The Initiator user group

To fully appreciate the user groups that were agreed upon in the end, the concept of the "Initiator" should be explained. In TelCorp., a person who started the administrative process after receiving a request to alter the telecommunications network signs their initials in a field called "originator". The significance of this is that this person could be from any of the families, or divisions, in TelCorp. The request to start the administrative process could be activated for many, many reasons.

The main reason, however, is when a request arises to connect a digital switch to an analog switch. This is a frequent occurrence. Everyone acknowledges that this is a period of transition for TelCorp. The implementation of digital technologies is changing the way people interrelate in the company. The roles of people in all of the "families" are changing. This fact caused a dilemma for both the developers and the TelCorp. Methods department. Should they assign this function to only one user group, who were the people who usually originated this administrative processes, or should they somehow define this permission to

reflect the variance of possibilities? The proposed solution was to create a new user group especially to initiate the administrative process.

This user group became known as the "Initiator". Anyone from any of the existing user groups could be a member of the Initiator user group. This indicated great flexibility for the concept of users groups and their relation to the network of nodes. The definition of user groups, until now, was based on the activities of the existing families. The fact that the Forms software could accommodate a more open approach to the definition of user groups indicated that the Administration software, in general, could not only mirror the existing administrative processes, but could accommodate change in the way user groups were defined.

Of all the user groups, the Initiator user group had the broadest range of access permissions. In the Forms Software User Guide, the Initiator had a large chapter devoted to the functions that it could perform.

The Forms Software User Guide

Both the organization and terminology of the *"Forms Software User Guide"* reflect an automated administration. The construct of the concept of administrative processes has been revised through the configuration of the Forms software to reflect the administrative processes of TelCorp. in an automated environment. The terms associated with the administrative processes have been reinterpreted to make sense in the automated administration of TelCorp.

Organization of the Forms Software User Guide

The construct of the concept of administrative processes has been revised through the configuration of the Forms. The revised construct of the administrative processes can be

illustrated in the organization of the Forms Software User Guide. Table X describes each of the chapters of the Forms Software User Guide in detail.

Table 2
Organization of this document

Chapter title	Description
Chapter 1, "General"	outlines the general information about this practice.
Chapter 2, "Introduction"	describes how Administration form processing is automated using the Forms software
Chapter 3, "Using Administration software procedures"	describes how to log on and use Administration software procedures
Chapter 4, "Getting started"	describes how to access and to use screens and softkeys
Chapter 5, "Working with the Admin Form 101"	includes procedures for manipulating the list of forms, opening, editing, routing, monitoring, and printing the Admin Form 101
Chapter 6, "Initiating Administration forms"	includes procedures for creating an Admin Form 101, filling in and forwarding an the Admin Form 101 to the next location.
Chapter 7, "Processing traffic assignments"	includes procedures for filling in av the Admin Form 101, for accessing traffic assignment functions, and forwarding the Admin Form 101 to the next location.
Chapter 8, "Processing transmission assignments"	includes procedures for filling in a the Admin Form 101, for accessing transmission assignment functions and forwarding the Admin Form 101 to the next location.
Chapter 9, "Controlling Administration forms"	includes procedures for filling in the Admin Form 101 and doing positive completion of the Admin Form 101.

The organization of this document is based on the functional needs of user groups. The functions for each user group are a collection of job-related tasks. A chapter is dedicated to working with Administration forms; a chapter deals with initiating Administration forms with the Forms software; and three chapters are dedicated to accessing traffic assignment, transmission assignment, and control functions with the Forms software. The names of these user groups, however, are the same as those existing in TelCorp. Each of these

names is considered a collection of job-related tasks. The Traffic and Transmission assigners have a chapter specifically written for them. The Chapter on Control can be used by anyone responsible for the control function. This was the OCB function, but as stated earlier, this function is expected to be decentralized among other job responsibilities. There are other user groups who do not have a chapter written specifically for them (Traffic monitoring and servicing and Transmission provisioning). These user groups are expected to perform the "Initiator" function. All user groups reference the same lexicon of Forms-software specific terms in the Forms Software User Guide.

Terminology

The terminology used in this document has been reinterpreted and revised to suit the automated Administration of TelCorp. Table 3 is included in a list of abbreviations and terms found throughout the document.

Table 3

Abbreviations and terms

Abbreviation or term	Definition
Archive	The archive is the last location on a Admin form 101 route.
Distribution list	A distribution list is a sequence of locations that the Admin form 101 follows in processing.
Location	A location is a position on a route. A location is also called a node.
Node	See the definition for location.
Path	See distribution list.
Profile	The form profile from which all Administration Forms 101 are created.
Route	A route is a series of locations that an Admin form 101 follows through the course of being processed. A route is also called a distribution list.
Template	The layout for the Administration Form 101.

This terminology can be traced back to the terminology of the GFT software. The appearance of these terms in the document confirms that this terminology made sense in the

context of the user community. The relevance of these terms was established through the conversations between the software designers and user representatives for 1990 through 1991. Through these conversations, the meaning of the concepts was clear and an understanding had been reached.

Built-in flexibility of the Forms software

All of the tasks associated with the configuration of nodes in Forms software are renegotiable. User permissions, the distribution path, and the definition of user groups can be modified at any time. This offers an amazing flexibility to TelCorp. The administration software can continue to change along with the structure, inter-departmental relations, administrative processes, administrative practices, policies, and use of information in the organization.

Seasonal summary

Through conversation, the Fall of 1991 revealed the following developments in the conceptualization of administration processes:

• The deadline for the production of the Forms software documentation was tight.

• The definition of the users permission changed radically up to the very last minute.

• The "Initiator" user group was created especially to originate the administrative process .

• The construct of the concept of administrative processes has been revised through the configuration of the Forms. The construct is reflected in the organization of the *"Forms Software User Guide"*.

• The GFT-specific terminology used in the *"Forms Software User Guide"* was reinterpreted and revised to suit the automated Administration of TelCorp.

• The tasks associated with the configuration of nodes in Forms software became renegotiable.

137

• The administration software can now continue to change along with the structure, inter-departmental relations, administrative processes, administrative practices, policies, and use of information in the organization.

Epilogue

Implementation strategy based on reinterpretation

All of the assumptions about software implementation that were implicit in the GFT software about the translation of a paper-based administration to an automated administration were confronted through conversation in the implementation of the Forms software at TelCorp. The following illustrates how each of the assumptions revealed in the section called "Translation of paper-based administration to an automated administration" in this chapter were resolved in the implementation of the Forms software at TelCorp.

Rather than directly replacing the paper-based administration by an automated administration as the original GFT documentation implies, the implementation strategy for the Forms software reinterpreted the paper-based administration to fit an appropriate method of organizing the TelCorp. administration in an automated environment. As such, the implementation strategy was based on administrative processes that accommodated the social-technical environment of an automated TelCorp. through conversation.

In this strategy, users' groups were defined as collections of job-related tasks. After a discussing the full ramifications of an automated administration in a changing organizational structure, the structure implied by a network of nodes provided an appropriate framework to automate the administration of TelCorp. After many, many meetings and discussions, job responsibilities were redefined as tasks. It was found that the structure implied by a network of nodes could accommodate changed in job responsibilities without disrupting the flow of administrative activities.

In this implementation strategy, the administrative practices of all user groups were changed to accommodate the use of a computer in administrative practices. Through

139

conversation, it was acknowledged that administrative practices could vary between locations based on the use of administrative information. The structure implied by a network of nodes could accommodate divergent administrative practices throughout the organization.

In this implementation strategy, the organizational structure was interpreted as a dynamic framework. Through conversation, it was discovered that the organizational structure was asymmetrical between regions. The structure implied through the network of nodes could accommodate variance in organizational structure once it had been identified.

In this implementation strategy, the interaction between families was recognized in pattern forms. Through conversation, it was revealed that the patterns of interaction between families were influenced by the people and histories within the departments. The structure implied by the network of nodes could accommodate these variable inter-departmental relations.

Talking back: the lesson learnt

Figure 10 illustrates that the social facts of the administrative processes were revealed through conversation between system designers and user representatives. The areas of knowledge that were addressed over time included: a) the structure of the organization; b) the support structure of the organization; c) the administrative activities of the organization; d) the use of information in the organization; e) the administrative practices of the organization; and f) the user community of the organization. In the end, the following social facts of the organization were inscribed in the Administration software: a) a dynamic organizational structure; b) variable inter-departmental relations; c) evolving job responsibilities; d) variable use of administrative information; e) divergent administrative

practices and f) changing user groups, defined as clusters of work-related tasks. The social

dynamic of the organization is indeed vibrant and in a state of flux.

Figure 10

Conversation in the software development process
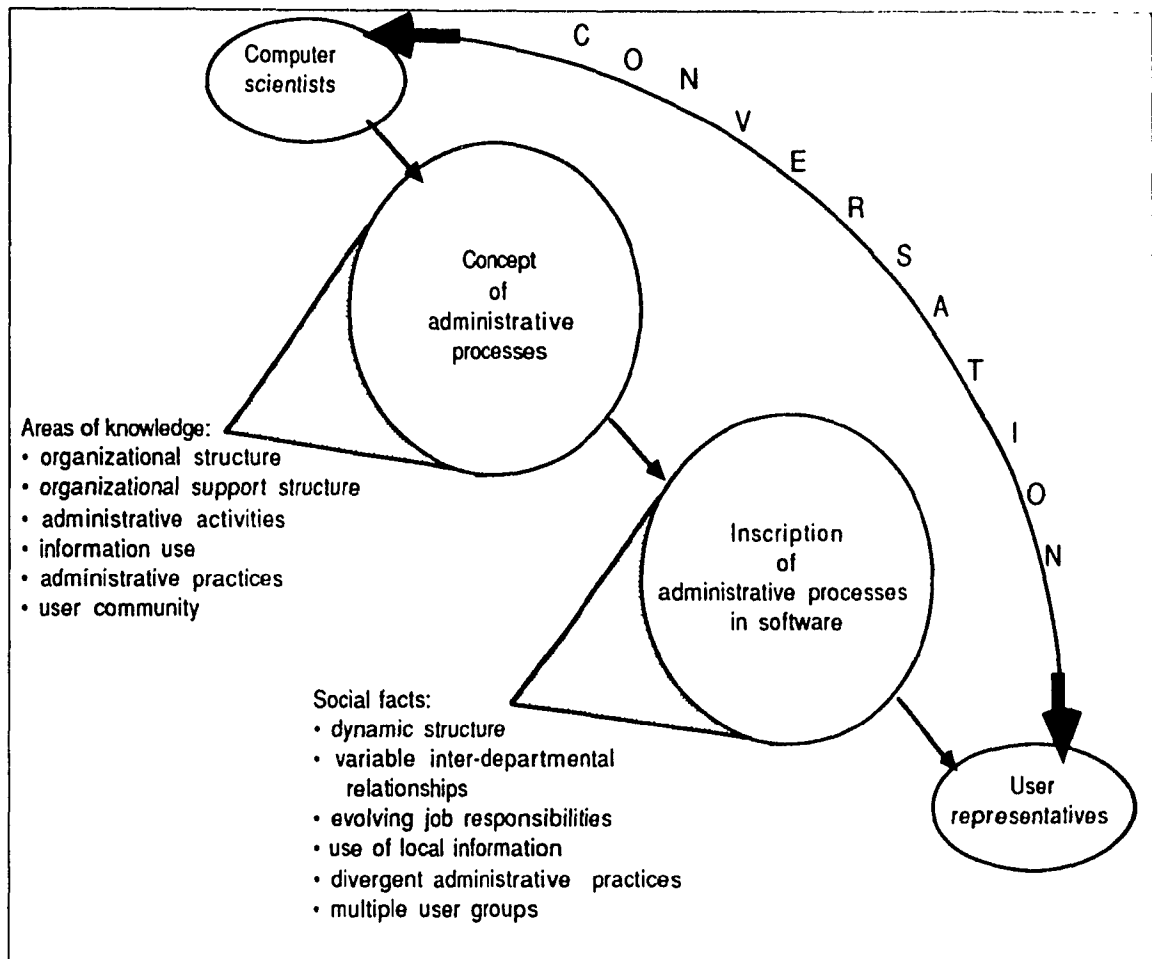
After delivering the Forms Software User Guide for the Field Trial, I moved over to

another project at TelCorp. I had been working with the Administration Software project

team for over two years. Although I developed bonds with the other team members, I felt

that a new project would rejuvenate me.

Now and then, I was privy to the developments of the project. The Field Trial was a

success. It is worthwhile mentioning, though, that the Administrative Form 101 continued

to change. Modifications were required each time the structure of the organization

changed, each time the inter-departmental relations changed, each time the job

responsibilities of a group changed, each time the use of information at a location changed,

each time local practices changed,or each time a new user group evolved in the network of

nodes.

Each of these changes indicated that the administrative processes were changing. The

social dynamics of the organization are dynamic. Each time the social dynamics of the

organization changes, the administrative processes adapt to the change. The social

dynamics of the organization made the administrative processes living processes,

constantly evolving.

The last I heard, Jeff B.'s boss was looking for someone to analyze the business

processes of the administration on an ongoing basis...

Chapter 4: Conclusions

Introduction

Through an examination of the objects of inquiry related to the problematic of this thesis, within the theoretical construct provided by a view of organizations as communicative patterns and current CSCW research that addresses how system designers and users interact, I have discovered several ways that the 'mutual learning' technique of systems analysis and design could be modified to incorporate the findings of this case study. The modifications of the systems analysis and systems design stages are a response to the following three major issues about the interactions between the software designers and users. First, there is a huge gap separating the perspectives of organizations held by computer scientists and users (or their representatives). Second, both need to understand the assumptions of these complementary perspectives to fully implement a system that reflects the organization in an automated environment. Finally, there are many barriers to communication between computer scientists and users (or their representatives) as a result of the practices of traditional computer science analysis and design.

In this thesis, I have tried to illustrate how the computer scientists construct their vision of an organization through the concept of administrative processes based on traditional methods of systems analysis and design. As a comparison, I have tried to show, how through conversation, the concepts of administrative processes were reinterpreted to fit into an automated organizational environment. I conclude that only through conversation between computer scientists and users (or their representatives), will systems be developed that accurately reflect the social dynamics of organizations.

Intention of this chapter

In this chapter I will summarize how the social dynamics of the organization were incorporated into the Forms software through conversation. Then, I will describe the role

of consensus in the definition of administrative processes. Third, I will present

recommendations for the traditional stages of systems analysis and systems design which

could be modified to incorporate the findings of this case study and to add to the mutual

learning approach to systems development. Finally, I will present a brief sketch of how

this case study could be extended to include the hypertext model of organization.

Incorporation of social dynamics into the concept of administrative practices

Through conversation, the social dynamics of the organization were incorporated into the

concept of administrative processes. The social dynamics of the organization were not

captured in the planning stage of software. This is not unusual. Traditional computer

science practices do not have a way of either capturing this information in their methods of

analyses, nor do they have a way of integrating this kind of information into their dominant

paradigm. The social dynamics of an organization come from knowledge that is outside the

realm of traditional framework for interpreting the computer scientists' world. This is

social knowledge, or knowledge about the social world.

The following areas of social knowledge were revealed through conversation in the

software development process: knowledge of the structure of the organization; knowledge

of the support structure of the organization; knowledge of administrative activities;

knowledge of the use of information within the organization, knowledge of the user

community of the organization. Through conversation, these areas of knowledge were

specified as social facts that could be implemented into the Forms software.

Role of conversation

Through conversation, the software developers were able to discover the social dynamics

of the organization. Through conversation, the user representatives were able to express

what changes in the concept of the administrative processes had to occur before the Forms

software, and the Administration software for that matter, could be fully implemented in a Field Trial. Not only did the software designers and the user representatives get to know each other better but these conversations enabled a discovery of social facts that could be inscribed in the software. These social facts included: a) a dynamic organizational structure; b) variable inter-departmental relations; c) evolving job responsibilities; d) the use of local information; e) divergent administrative practices; and f) multiple user groups.

Through conversation, the users were able to find a mechanism to arrive at a consensus about what the administrative processes were.

Role of consensus in defining administrative processes

The conversations between user representatives were indispensable in the software development process. Otherwise, the concept of administrative processes could not have been defined in a way that reflected a shared view of administrative processes. This is important because the administration is at the centre of the organization. Administration requires a lot of coordination and communication between people. In fact, administration is little else than the coordination and communication between the employees at TelCorp.

Coordination and communication breaks down as a result of misunderstandings. Often, misunderstandings create confusion and ambiguity about 'what is to be done'. In the paper-based administrative system,. the Administration Form 101 was passed through the channels and sent back and forth between departments until 'the right things had been done'. Or, all of the people involved in the sign-off of that form were satisfied with the activities that had been performed. Also, because it was paper-based, people could literally write anything on it, regardless of the layout of the form. In situations that arose where the form couldn't fulfill the specific needs of the information required for that particular situation, this is exactly what happened.

145

The versatility of the paper medium fulfilled many purposes in the administration processes. As paper, it could adapt to the administrative situation readily. The end result was that the completion of the forms, indicated an approval of the administrative activities that had occurred throughout the administrative processes. This was a stamp of integrity of the administrative processes. In the paper-based administrative system, consensus was required between the families of TelCorp. In the paper-based system, consensus emerged in the coordination of these administrative processes through the channel of the Administration Form 101.

The effect of automation, in this case study, was to elevate the need for consensus between the families of TelCorp. about the integrity of the administrative processes to face-to-face conversations. This was why it was so important that the user representatives arrived at consensus building as a review process.

Mutual learning approach to systems analysis

As stated previously, Morten King has coined the phrase 'mutual learning' to replace systems analysis. Mutual learning implies that designers learn about the application area and users learn about new technical possibilities.

In the case that I have presented, mutual learning took place during the implementation phase. This is true insofar as the system designers learnt about the administrative processes and the user representatives learned how to configure the Forms software. However, the impact of conversation on the configuration of Forms software (i.e. the inscription of the social facts) cannot be described *only* by mutual learning. Not only did the system designers and user representatives learn about each other's working world, but through conversation, they established a way of sharing their worlds.

The inscription of social facts into the Forms software represented a shared view of administrative processes. This view was shared by the user representatives, the software designers.

In this conclusion, I propose that the research of CSCW could benefit by incorporating the findings of the impact of conversation in the software development process into their concept of mutual learning.

The following section makes recommendations about how the traditional stages of systems analysis and systems design could be modified to incorporate the findings of this case study. These recommendations are made above and beyond the tasks that are normally associated with these stages. My intention is not to rewrite the entire software development process, but to add to the mutual learning approach to systems development.

Recommendations for the system analysis and design stages

The following recommendations are made with reference to an automated administration project.

Systems analysis stage

1—Make the following assumptions during this stage:

    a) Assume that the organizational structure is dynamic.

    b) Assume that departmental relationships are based on the people and histories of the departments.

    c) Assume that job responsibilities are evolving.

    d) Assume that information is used locally.

    e) Assume that administrative practices are local.

    f) Assume that the user groups are based on a collection of job-related tasks.

2—Use traditional formal analyses to establish information needs of the organization. This technique will result in the functional analysis of the administration where each function represented a process to administrative information.

3—Do not confuse information processes with administrative processes.

4—Assemble a software project team consisting of members from software development and people who can represent the diversity of users. Establish the following with this team:

    a) What their interests are.

    b) What their areas of expertise are (ie. background knowledge).

    c) Establish a mechanism for the team to arrive at a consensus.

    d) Set up regular times to meet.

5—Conduct analyses of social dynamics of the organization *early*.

6—The following types of analyses should be performed:

    a) Conduct studies of anticipated changes to the organizational structure.

    b) Conduct task analyses of jobs. (Don't assume that official administrative practices are how people actually work.) Analysis should incorporate exceptions. Special attention should be paid to how people handle exceptional circumstances.

    c) Conduct studies of how people use information locally. Use the bubble diagram technique to analyze the business (administrative) activities. Divide the organization into natural groupings and conduct local studies. Then compare locations to find patterns and account for differences.

Systems design stage

7—Base design on the requirements of local administrative processes.

8—Do not map local administrative requirements to the system specifications (as was done in the section called "Mapping of business functions to the Generic Forms Tool (GFT) functions" in *Chapter 2: analytic foundation*). Rather, use local information requirements from which to identify nodes. Then, build design from the required administrative tasks associated with each node.

9—Ensure that any problems within the dynamics of the software project team are addressed. Maintain open communication between team members.

10— Create a Project Management Plan that includes the definition of nodes, user groups, and distribution lists (configuration) early in the sequence of the development of software features.

Role of social scientist in system development process

A social scientist could greatly assists the recommendations listed above. The role of a social scientist could perform the following :

   a) act as facilitator to the software project team

   b) conduct studies based on the participant-observer technique in the client's organization

   c) present the following research results to all team members:

      - anticipated changes to organizational structure

      - use of local information

      - task analysis of job responsibilities

      - current evolution of organizational structure

      - identify preliminary user groups

Bibliography

Berger and Luckmann (1967). The social construction of reality. NY.: Anchor Books.

Burgess (1984). In the field. London: Allen & Unxin Ltd.

Chalmers, R.(1983). Rural development. Essex, England: Longman Scientific &
Technical.

Child , Ganter, & Kiesler. "Technological innovation and organizational conservatism".
In J.M.Jennings and A. Bertendam (Ed.), New technology as organizational innovation.
London: Bollinger, Chap. 5, pp. 87-115.

Gouldner, A. (1976). The dialectic of ideology and technology. New York: Oxford
University Press.

Francik, Rudman, Cooper, Levine "Putting Innovation to work: Adoption Strategies for
multimedia communication systems". Communications of the ACM, December 1991, pp.
53-63.

Foucault, M. (1976).The archaelogy of knowledge. New York: Harper & Row.

Gardner, H. (1985). The mind's new science. New York: Basic Books.

Garfinkel, H. and Sacks, H. (1986). "On Formal Structures of Practical Actions". In
Ethnomethodological studies of work. London: Routledge & Kegan Paul.

Halliday, M. and Hasan, R. (1986). <u>Language, context, and text: Aspects of a language in a social-semiotic perspective</u>. England: Deakin University.

Handel, W. (1982). <u>How people make sense</u>. New Jersey: Prentice-Hall.

Heritage, J. (1984). <u>Garfinkel & Ethonomethodology</u>. Cambridge: Polity Press.

Hodge and Kress (1988). <u>Social Semiotics</u>. England: Cornell University Press.

Huberman, Tessler & Cameron. "A Generic Forms Handling Tool and its Application to Network Trouble Management". <u>Proceedings from ICC '89</u>, Boston, June 1989.

Kraemer & King. "Computing and public organizations". <u>Public Administration Review</u>, Volume 46, 1986, pp. 488-496

Kling, R. "Cooperation, Coordination and Control in Computer-Supported Work". <u>Communications of the ACM</u>, December 1991, pp. 83-88.

Kling, R. "Social analyses of computing: Theoretical perspectives in recent empirical research". <u>Computing Surveys</u>, Volume 12, March 1980, pp. 61-110.

Kyng, M. "Designing for Cooperation: Cooperating in Design". <u>Communications of the ACM</u>, December 1991, pp.65-73.

Lucas, H. "Utilizing information technology: Guidelines for managers". <u>Sloan Management Review</u>, Fall 1986, pp.39-47.

Markus, M. & Robey, D. "Information tehcnology and organizational change: Causal structure in theory and research". <u>Management Science</u>, Volume 35, Number 5, May 1988, pp. 583-598.

Mandell, S. (1982). <u>Computers and Data Processing</u>. Minnesota: West Publishing Co.

March & Simon (1958). <u>Organizations</u>. New York: John Wiley.

Moerman, M. (1988). <u>Talking culture, ethnography and conversation analysis.</u> Philidalphia: University of Pennsylvania Press.

Mohrman, A. & Lawler, W. (1984). "A review of theory and research". In F. Warren & N.C. Larlan (Ed.), <u>The information systems research challenge</u>. Boston: Harvard Business School Press.

Morgan, G. (1986). <u>Images of organization</u>. California: Sage.

Morieux, Y. & Sutherland, E. "The interaction between the use of information technology and organizational culture". <u>Behaviour and Information Technology</u>, 1988, Volume 7, Number 2, pp. 205-213.

Norman, D. "Collaborative Computing: Collaboration First, Computing Second". <u>Communications of the ACM</u>, December 1991, pp. 88-90.

Olson, M. & Lucas, H. "The impact of office automation on the organization: Some implications for research and practice". Communications of the ACM, November 1982, pp. 838-847

Perrow, C. (1986). Complex organizations. New York: Random House Inc.

Quinn, J. & Paquette, P. "Technology in services: Creating organizational revolutions". Sloan Management Review, Winter 1990, pp. 67-78.

Robey, D. "Computer information systems and organization structure". Communications of the ACM, October 1981, pp. 679-687.

Robey, D. "Computers and management structure: Some empirical findings re-examined". Human Relations, Volume 30, Number 11, 1977, pp. 963-976.

Smircich, L. Concepts of culture and organizational analysis, Administrative Science Quarterly, 1983.

Suchman, L. (1987). Plans and Situated Action: The problem of human/machine interaction. Cambridge & New York: Cambridge University Press.

Taylor, J. (Forthcoming). The vulnerable fortress.

Webster's New World Dictionary, Toronto: Nelson, Foster & Scott Ltd. pp. 18.

Weick, K. (1976). "Educational Organizations As Loosely Coupled Systems".

Administrative Science Quarterly 21 (March), pp. 1-19.

Weick, K. (1979). The social psychology of organizing. New York: Random House.

Weick, K. (1983). "Research Agenda." In L. Putman and M. Pacanowsky (Ed.),

Communication and Organization an Interpretive Approach (pp.13-31).San Francisco:

Sage.

Zuboff, S. (1984). In the age of the smart machine. New York: Basic Books.

Corporate documents

R&D Tel. Operations plan. 1986.

R&D Tel. Technical proposal. 1988.

TelCorp. User Needs Requirements 1989.

R&D Tel. Project Management Plan. 1989.

R&D Tel. GFT Overview. 1989.

Appendix A

Future research

The recommendations included in this chapter, as well as the framework for this case study, are based on the writings and teachings of J. Taylor. In The Vulnerable Fortress (Forthcoming), Taylor traces the history and effects of new communications technology in organizations. For Taylor, an understanding of the new communications technologies is fundamental to our understanding of organizational growth, human relations, and ultimately, the way we govern ourselves through institutions. New communications technologies have altered the perceptions, policies, and practices of organizations.

As an extended case study of the implementation of communications technology, the findings of this case can be used in future research of the modelling of organizations. Current research in the use of models for organizational studies suggests that the hypertext model can be used effectively to understand the effect of communications technologies in organizations (Taylor 1991). The results of this case can be used as empirical evidence of how the implementation of communications technology can be interpreted accurately with a hypertext model of organizations.

In studying the effect of communication technologies in organizations, Taylor suggests that we examine our organizational models.

> "The implementation of new communication and information technologies over the past generation has, I claim, produced the kind of anomaly that emerges when the reality of organizational process is at variance with our concepts of it, in that the models of organization we are accustomed to rely on no longer describe very well the communicational reality they are supposed to represent."[31]

---

[31] Taylor, J., The Vulnerable Fortress (Forthcoming), pp.6.

Taylor presents the hypertext model as a way of understanding the current changes of organizations. This is not the place for a full description of the hypertext model of organizations. The following is a brief quote from The Vulnerable Fortress, "A hypertext theory would visualize organization not as a fixed, immutable structure but as a set of alternative possible transactional arrangements. Organization then becomes the *structure emerging out of the process* of communication, just as communication is the process by which the structure of organization progressively manifests itself. To the communications theorist, to talk about THE organization is to miss the point: all organizations are filtered through the perceptive systems of the people who experience them and for whom they provide a backdrop of reality. Each perception of the organization is as valid as any other; to the scientist there is no "official version" and no way to discover the "real" truth about the organization other than through the filtered perceptions of the people for whom it is a reality. And there are as many perceptions (however frequently they converge) as there are people. We therefore ask of a model of organization that it incorporates these properties, of being open-ended, multi-faceted, and yet visibly structured, and indeed hierarchically structured."[32]

The hypertext model provided an excellent model for the interpretation of this case study. Due to time limitations, I was not able to apply these research findings to build upon the hypertext model or organizations. This is an area for future research.

---

[32] Ibid, pp. 120

Future research design

Problematic:

How the hypertext model can be extended to accommodate a diversity of organizations in a broad spectrum of industries.

Methodology:

A series of case studies.

Assumptions:

The recommendations that are proposed in this chapter would be used as a guideline in which to structure the participant-observer methodology.

Rationale:

Aside from this unique approach, this research would benefit all the organizations that agreed to participate. The role of the social scientist in these projects would not only facilitate the management of the projects but would install mechanisms in their software development processes that could be applied to other projects in their organizations.

Moreover, future research would broaden the nature, and applicability, of Computer Supported Cooperative Work. This research would supplement existing empirical data about cooperative work from actual organizations.

Finally, this research would extend the hypertext model into the area of organizational and communications studies.