

1983

User Developed Computer-based Applications: A Model Of The Factors Of Success

Suzanne Rivard

Follow this and additional works at: <https://ir.lib.uwo.ca/digitizedtheses>

Recommended Citation

Rivard, Suzanne, "User Developed Computer-based Applications: A Model Of The Factors Of Success" (1983). *Digitized Theses*. 1256. <https://ir.lib.uwo.ca/digitizedtheses/1256>

This Dissertation is brought to you for free and open access by the Digitized Special Collections at Scholarship@Western. It has been accepted for inclusion in Digitized Theses by an authorized administrator of Scholarship@Western. For more information, please contact tadam@uwo.ca, wlsadmin@uwo.ca.

The author of this thesis has granted The University of Western Ontario a non-exclusive license to reproduce and distribute copies of this thesis to users of Western Libraries. Copyright remains with the author.

Electronic theses and dissertations available in The University of Western Ontario's institutional repository (Scholarship@Western) are solely for the purpose of private study and research. They may not be copied or reproduced, except as permitted by copyright laws, without written authority of the copyright owner. Any commercial use or publication is strictly prohibited.

The original copyright license attesting to these terms and signed by the author of this thesis may be found in the original print version of the thesis, held by Western Libraries.

The thesis approval page signed by the examining committee may also be found in the original print version of the thesis held in Western Libraries.

Please contact Western Libraries for further information:

E-mail: libadmin@uwo.ca

Telephone: (519) 661-2111 Ext. 84796

Web site: <http://www.lib.uwo.ca/>

CANADIAN THESES ON MICROFICHE

S.B.N.

THESES CANADIENNES SUR MICROFICHE



National Library of Canada
Collections Development Branch

Canadian Theses on
Microfiche Service

Ottawa, Canada
K1A 0N4

Bibliothèque nationale du Canada
Direction du développement des collections

Service des thèses canadiennes
sur microfiche

NOTICE

AVIS

The quality of this microfiche 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 a poor photocopy.

Previously copyrighted materials (journal articles, published tests, etc.) are not filmed.

Reproduction in full or in part of this film is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30. Please read the authorization forms which accompany this thesis.

La qualité de cette microfiche dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

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

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

Les documents qui font déjà l'objet d'un droit d'auteur (articles de revue, examens publiés, etc.) ne sont pas microfilmés.

La reproduction, même partielle, de ce microfilm est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30. Veuillez prendre connaissance des formules d'autorisation qui accompagnent cette thèse.

THIS DISSERTATION
HAS BEEN MICROFILMED
EXACTLY AS RECEIVED

LA THÈSE A ÉTÉ
MICROFILMÉE TELLE QUE
NOUS L'AVONS REÇUE

USER DEVELOPED COMPUTER-BASED APPLICATIONS:
A MODEL OF THE FACTORS OF SUCCESS

VOLUME I

by

Suzanne Rivard

School of Business Administration

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Ontario
December, 1982

© Suzanne Rivard 1982.

ABSTRACT

The development of computer-based applications by non-Data Processing people in organizations is a relatively new phenomenon, which is said to have the potential of solving some of the problems inherent in application development by Data Processing (DP) professionals. First, user developed computer-based applications (UDA) is said to relieve DP application development and maintenance workload. Second, by eliminating the intermediary role played by the systems designer, UDA is said to eliminate communication problems which often lead to difficulties in the development process.

While UDA is spreading rapidly, little research has been done on it, and organizations considering its introduction face the issue of what to do so as to make it a successful endeavor. The purpose of this research was to develop and provide empirical evidence for a model of the factors of success for UDA.

The study was conducted in phases and followed a multi-method approach. An initial research model was first developed, which integrated the UDA literature with literature from other relevant areas. This model served as a basis for the field investigation, wherein interviews (structured and unstructured) were conducted in 10 large firms, with DP managers, DP professionals providing UDA support, and users. The outcomes of this phase were a description of UDA in each firm studied, and a revised, improved version of the research model. This revised version consists of two sub-models. The Favorable

model. This revised version consists of two sub-models. The Favorable Cost Benefits of UDA Sub-Model stresses the importance, for DP, of the UDA evaluation process. The User Satisfaction Sub-Model defines success from a user point of view.

The survey phase which followed the field investigation was aimed at obtaining additional data on UDA from a user point of view, and at testing the relationships embodied in the User Satisfaction Sub-Model.

Data analysis, based on the questionnaire responses of 272 users, provided support for nine of the 10 propositions in the model. It was found that user satisfaction with independence from DP, user satisfaction with support provided by DP, user friendliness of the software tools, user attitude toward UDA, and DP readiness for change, were the most important variables in explaining overall user satisfaction with UDA.

While the results obtained provide support for the model, more research is needed to improve knowledge and understanding of UDA. Areas for future research include (1) the broadening of the sampling frame so as to study situations wherein DP does not provide UDA tools and support to users; (2) the investigation into the economics of UDA; and (3) the study of the role of Data Administration in the context of UDA.

ACKNOWLEDGEMENTS

Many people contributed to the generation of this thesis, and I wish to express my gratitude to them.

My undertaking of doctoral studies would have been impossible without the generous financial support provided by the Ecole des Hautes Etudes Commerciales, Université de Montréal. Funds for conducting this study were made available through a School of Business Plan for Excellence Doctoral Program Research Grant, and by the Direction de la Recherche, Ecole des Hautes Etudes Commerciales.

I am deeply indebted to my thesis advisor, Professor Sid Huff, for providing the impetus for conducting this study. His thoroughness and insight, as well as the enthusiasm he showed were the best support a doctoral student can ever receive.

Among my colleagues, I am particularly grateful to Dan Gardiner and Jean Talbot. Both willingly accepted my frequently interrupting their own work for discussing my research.

To my husband, Roger, I owe more than I can ever repay. Roger directly contributed to my study, by coding and entering the questionnaire data, typing most of the thesis, and proof-reading it many times. More importantly though, he volunteered for taking the responsibility of caring for our daughter, and provided a home wherein optimism reigned.

To Marie-Frédérique, who provided balance to my life, I
dedicate this thesis.

TABLE OF CONTENTS

CERTIFICATE OF EXAMINATION.....	ii
ABSTRACT.....	iii
ACKNOWLEDGEMENTS.....	v
TABLE OF CONTENTS.....	vii
LIST OF TABLES.....	ix
LIST OF FIGURES.....	xi
CHAPTER I - FOCUSING THE RESEARCH	
1.1 Introduction and Research Objectives.....	1
1.2 Key Concepts.....	4
1.3 Organization of the Thesis.....	7
CHAPTER II - LITERATURE REVIEW	
2.1 Origin of UDA.....	9
2.2 UDA Literature.....	24
2.3 Other Areas Relevant to UDA.....	60
2.4 Summary.....	90
Footnotes.....	91
CHAPTER III - INITIAL RESEARCH MODEL	
3.1 The Dependent Variable.....	95
3.2 The Independent Variables.....	101
3.3 Relationships Among the Variables.....	111
3.4 Summary.....	115
Footnotes.....	116
CHAPTER IV - RESEARCH METHODOLOGY	
4.1 Choice of an Approach.....	117
4.2 The Three Phases of the Study.....	119
4.3 Summary.....	126
Footnotes.....	127
CHAPTER V - PRESENTATION AND ANALYSIS OF THE INTERVIEW DATA	
5.1 Background Information on the Firms Studied.....	129
5.2 Evaluation of the Initial Research Model.....	157
5.4 A Revised Version of the Research Model.....	183
5.4 Summing Up the Interview Phase.....	193
Footnotes.....	195
CHAPTER VI - PRESENTATION AND ANALYSIS OF THE QUESTIONNAIRE DATA	
6.1 Methodological Aspects of Phase III.....	197
6.2 A First Look at the Data.....	206
6.3 Testing the Model.....	256
6.4 Summing Up the Survey Phase.....	267
Footnotes.....	269
CHAPTER VII - TAKING STOCK OF THE STUDY	
7.1 Summing Up the Research Process.....	271
7.2 Managerial Implications of the Research Findings...	284

7.3 Implications for MIS Research.....	289
7.4 Conclusion.....	295
* * *	
APPENDIX A. QUESTIONNAIRE AND LETTER: PHASE I.....	296
APPENDIX B. LETTER, EXECUTIVE SUMMARY, AND CHECK-LIST FOR DP INTERVIEWS.....	302
APPENDIX C. DP QUESTIONNAIRE.....	309
APPENDIX D. CHECK-LIST FOR USER INTERVIEWS.....	332
APPENDIX E. SURVEY QUESTIONNAIRE.....	335
APPENDIX F. SCENARIOS OF UDA.....	352
APPENDIX G. PROGRAMMING TOOLS.....	462
APPENDIX H. REMINDER.....	468
APPENDIX I. COMPOSITE SCALES.....	470
BIBLIOGRAPHY.....	477
VITA.....	488

LIST OF TABLES

Table	Description	Page
2.1	Installed Systems vs. Backlog	14
2.2	Breakdown of Maintenance Activities	15
2.3	Models of Change	87
3.1	Independent Variables	95
4.1	Results of the Identification Phase	120
4.2	Industrial Breakdown for Participating Firms	120
4.3	Response Rates	126
5.1	The Companies	131
5.2	Overview of DP in the Firms Studied	133
5.3	DP Side of UDA	135
5.4	Tools in the Firms Studied	139
5.5	Proportion of Interviewees per Taxonomy Category	147
5.6	DP Definition of UDA Success	162
5.7	DP Management Concerns with Regard to UDA	176
6.1	Number of Questionnaires Distributed per Firm	201
6.2	Response Rate per Firm	203
6.3	Response Rates	204
6.4	Distribution of Respondents Across Firms	205
6.5	Distribution of Respondents per Functional Area	208
6.6	Distribution of Respondents per Job Category	208
6.7	Variables Used to Classify Respondents	211
6.8	Comparison of the Three Categories of Users	212
6.9	Results of the First SPSS Classification of Users	215
6.10	Results of SPSS Second Classification	217
6.11	Distribution of Respondents per Taxonomy Category	217
6.12	Summary of Analyses for: Characteristics of Users Who Develop Applications	219
6.13	Comparisons Across User Categories	221
6.14	User Category vs. Application Development for Others	222
6.15	User Category vs. Formal Duty	223
6.16	User Category vs. Job Prior to UDA	225
6.17	Computer Background of Users	228
6.18	Percentage of UDA Time per Organizational Entity Supported	231
6.19	Periodicity of Use of Applications	236
6.20	Application Development Mode	237
6.21	Average Percentage of UDA Activities per Computer Type	237
6.22	Average Percentage of UDA Activities per Terminal Location	239

6.23	Programming Tools Used	240
6.24	Major Advantages of UDA	244
6.25	Major Disadvantages of UDA	246
6.26	Productivity Improvement	248
6.27	Aspects of UDA Evaluated by Users	249
6.28	Summary of Results of Proposition Testing	259
6.29	Regression Model	266

LIST OF FIGURES

Figure	Description	Page
3.1	Initial Research Model	111
5.1	Initial Research Model	158
5.2	Success of UDA, as Seen by DP	165
5.3	"User Satisfaction" Sub-Model	184
5.4	"Favorable Cost Benefits of UDA" Sub-Model	189
6.1	Information Usage	226
6.2	Percentage of UDA Time Spent per UDA Activity	230
6.3	Time Required for Development	233
6.4	Source of Data for Applications	234
6.5	Who runs the Applications	242
6.6	Importance vs. Satisfaction	250

CHAPTER I
FOCUSING THE RESEARCH

1.1 INTRODUCTION AND RESEARCH OBJECTIVES

Development of computer-based applications receives much attention in the MIS field. A recent Datamation special report claims that "application development remains one of the Data Processing (DP) industry's thorniest problems" (Runyan, 1981, p.165). Problems such as large backlogs of applications, projects months and years behind schedule, shortage of programmers, applications which do not respond to user needs in a satisfactory way have generated much search for solutions.

Elements of solution to the application development problem come from three main directions. First, Modern Programming Techniques such as structured walk-throughs, chief programmer team, automatic flowcharters, and so on, are claimed to have the potential of increasing programmer productivity (Daly, 1979). Second, new approaches for eliciting user information requirements are hoped to provide systems which better fit user needs (Cooper and Swanson, 1979). Finally, new systems development methodologies have been introduced. Such methodologies are based on the assumption that users have to be actively involved in the development of a system, the DP professionals playing the role of consultants (Rudawitz and Freeman, 1981; Mumford, 1981).

Recently, several authors (McLean, 1979; McCracken, 1980; Martin, 1982) have suggested that applications developed by end users themselves is a promising element of solution to the application development problem, by relieving DP professionals of some of their workload, as well as by eliminating some of the communications problems between users and DP professionals. Moreover, in the report of their study on the growth potential of the Data Processing industry for the period of 1980-1985, Dolotta, Bernstein, Diekson, et al. (1976) claim that:

As the demand for larger and more complex applications systems continues to exceed the supply, this [applications developed by end users] appears to be the only way, barring some startling breakthrough, in which the growth of the data processing industry can continue to be sustained. (Dolotta et al., 1976, p.113)

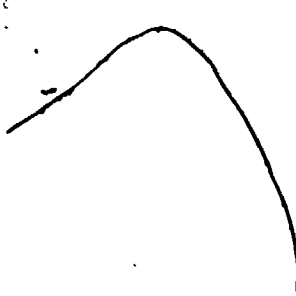
User developed computer-based applications (UDA) is still a relatively new phenomenon. According to Rockart and Flannery (1981), most companies are just beginning to make application development tools available to their users. However, they further suggest that UDA is spreading rapidly.

Because of the newness of the phenomenon, little is known about the circumstances under which an organization should consider introducing UDA, or about what should be done to make it a successful endeavor. Some authors (McLean, 1979; McCracken, 1980; Martin, 1982) identify variables which they believe are critical to success of UDA. However, little empirical data exist to support (or disprove)

statements about the importance of such variables, or to identify other important variables.

The purpose of this study was to develop and provide empirical evidence for a model of the factors of success for UDA. The study investigated situations where end users developed administrative, computer-based applications either for themselves or for other end users. The research approach chosen for conducting the study put as much emphasis on theory generating as on theory testing.

An initial research model was first developed. The model was based on the literature on UDA as well as on literature on other relevant areas. Structured and unstructured interviews were then conducted with DP people and with users in 10 large Canadian firms where UDA existed. This interview phase was undertaken in order to better understand UDA and its organizational context, and to submit the initial research model to the "acid test" of the real world. The main outcomes of this phase of the study are (1) a description of UDA in the organizations studied, including a user taxonomy, and (2) a revised, improved version of the research model. Finally, a survey was conducted so as to obtain additional descriptive information on UDA from a large number of users, and to test some of the relationships implied by the research model, as revised during the interview phase.



1.2 KEY CONCEPTS

To insure that the terminology used in this study is clear, two key concepts are defined and discussed below.

1.2.1 Users

The MIS literature provides a broad range of meanings for the term users. From a vendor's point of view, users is often a synonym for customers and means people in DP departments. Some people in the field define users as those who receive information from a computer-based application, while others see the users as those who feed the data into a system. Taxonomies of the user population are found in the literature (see Dolotta et al., 1976; CODASYL, 1979; McLean, 1979). The present study uses the taxonomy suggested by Dolotta et al. to identify the category of users who will be studied.

Dolotta et al. define three categories of users: end users, mid users, and system support users. These differ in their role, their knowledge, and their expectations in the context of DP. System support users are responsible for providing the other two categories of users with the "basic data processing environment." They have a very good knowledge of the technical aspects of DP and they are concerned with efficiency of hardware and software. Systems programmers and systems engineers fall in this category.

The mid users could also be called systems developers; "they are those who interpret the end users' needs and develop the procedures a data system applies to obtain the solutions to the problems posed by the end users" (Dolotta et al., 1976, p.41). Mid users have good knowledge of DP and they expect to evolve in an environment which provides them with the tools they need to perform their roles, without having to be concerned with internal details of the computer system. Systems analysts and applications programmers are mid users.

The end users are the individuals for whom DP is a tool which is used to assist them in performing their job, by providing them with information, processing transactions, or performing calculations. End users are members of departments other than DP. They usually do not have much knowledge of DP, as compared to users in the other two categories. End users are further differentiated according to the degree to which they are involved with the development and the use of an application. Indirect end users receive information from an application. Intermediate end users play a role in the development of an application in working with the mid user to specify the problem a system has to solve. Direct end users can be those who routinely use a system in their daily tasks. They can also be of a more active type, when they "conceive the problem, formulate the solution, and use the system to obtain the solution." (Dolotta et al., p.44)

The target group of this study are those end users who take on the role of mid users in the development of applications. These individuals may or may not be direct users of the applications they

develop.

1.2.2 Computer-Based Applications

For the purposes of this study the use of the phrase computer-based applications is not restricted to transaction processing systems, but encompasses all of the following.

- a. Computer-based models, the parameters of which are changed to answer "what if" type of questions.
- b. Programs which process transactions and update files.
- c. Programs which retrieve data from a file or a data base, perform calculations on the data, and produce reports.
- d. Programs which retrieve data from a file or a data base and produce a list or report, without performing any calculations.
- e. Programs which perform data analysis (e.g. preparing contingency tables, performing regression analysis, etc.).
- f. Programs which produce graphics, using graphics packages.

A program is further defined as being a set of instructions which a computer can be made to execute. Those instructions may be written using general purpose programming languages, statistical packages, data base query languages, modeling languages, report generators, etc.

1.3 ORGANIZATION OF THE THESIS

Chapter II reviews the literature in order to map the terrain and to establish the premises on which to build an initial research model. Chapter III presents the initial research model which is both an attempt to integrate the literature on UDA with the literature of other relevant areas, and a research tool to be used as a basis for investigation. Chapter IV discusses the research methodology. The study was conducted in phases and adopted a multi-method approach. Chapter V presents and discusses the results of the interview phase of the study, while Chapter VI focuses on the results of the survey phase. Finally, Chapter VII takes stock of the present study, by summing up the research results and discussing implications of the study for MIS managers and researchers.

The appendices include the research instruments which were used during the investigation phase of this study (Appendices A, B, C, D, E, and H), scenarios of UDA in each of the 10 organizations studied (Appendix F), a description of the major programming tools used in those organizations (Appendix G), and a discussion of the composite scales used in the survey questionnaire (Appendix I).

NOTE: Footnotes are indicated by square brackets [] and are to be found at the end of each chapter.

CHAPTER II

LITERATURE REVIEW

Literature directly pertaining to UDA is relatively scarce, and up to now is mainly found in trade journals as opposed to scholarly publications. However, the concept of UDA is related to several other areas of MIS: software development productivity, centralization decentralization of DP, and data administration to name a few. This chapter reviews both the UDA literature as well as the literature of other areas that are relevant to the present study.

The chapter is organized as follows. Section 2.1 discusses the phenomena from which UDA originates; these phenomena are classified in two categories, namely, DP problems in the context of systems development and users' problems in the context of systems development. Section 2.2 reviews the UDA literature. Literature on UDA appears to naturally fall into one of the following three categories: essays on UDA, reported experience of firms with UDA, and research-based literature. Finally, the third section gives an overview of the literature in other relevant areas: centralization vs. decentralization of DP, data administration, human-computer interactions, and organization change.

2.1 ORIGIN OF UDA

The idea of having users develop their own applications is not entirely new. For instance, in 1969, Sammet suggested two approaches for increasing programmer productivity. The first is to provide programmers with better tools with which they can perform their work. The second is to "make everybody his own programmer, just as almost everybody drives his own car rather than obtaining the services of a professional driver" (Sammet, 1969, p.730). Note that the latter is a means of relieving DP professionals of some of their workload rather than a means of increasing their productivity. Boehm (1973) sees a trend toward the day where most problems in an area could be programmed by an end user "in less than an hour...with one day of specialized training." (Boehm, 1973, p.50)

This section discusses phenomena which created the need for UDA. These phenomena are grouped under two headings: DP problems in the context of systems development and users' problems in the context of systems development. Finally, the emergence of UDA, in conjunction with the phenomena that made it feasible, is discussed.

2.1.1 DP Problems in the Context of Systems Development

Several of the problems encountered by the DP departments in the context of application development are related to the issue of DP professionals' productivity. Such problems include large backlogs of applications, projects behind schedule, and the resulting user

dissatisfaction.

For illustrative purposes let us look at a DP department and the requests for application development and maintenance as a queuing system. The DP department, at a given point in time provides the user community with a number of servers N . Once a request has been accepted, it has to enter the queue. A major problem is that the queue length and the waiting time are such that the customers (users) are dissatisfied. As a result, users may decide to stay in the queue (yet, being very dissatisfied), leave the queue to go somewhere else to be served (if they have the option to go elsewhere), or they may simply leave the queue and do without their application. However, no matter what the users decide to do, DP's reputation of service is likely to suffer.

One means of reducing the waiting time (given that the arrival rate and the number of servers remain constant) is to reduce the service time, that is, to increase the servers' productivity. During the past decade several tools have been put into the hands of programmers in order to increase productivity. Sackman, Erickson, and Grant (1968) for instance, conducted an experiment in order to compare batch versus on-line programming. The authors claim that a median improvement of 20 percent is obtained by using on-line programming. However, in their study, individual differences accounted for much more differences in productivity. The difference between best and worst programmers for the problem used in the experiment was as high as 27 to one for debug hours and 20 to one for coding hours. An important

conclusion one can draw from this experiment is that hiring capable programmers is a better way of increasing software development productivity than on-line programming.

In addition to on-line programming, several "modern programming practices" were introduced. Jones (1979) claims that in the past decade:

Over a hundred new technical methods have been introduced, and programming has moved several paces along the path of becoming "software engineering" instead of a more or less artistic endeavor. (Jones, 1979, p.79)

These new methods include techniques such as structured programming, structured walk-throughs, programmer team operation, automatic flow charters, test case generators, use of a program development support library, and so on (Holton, 1977; Lehman, 1979). While some claim that such modern practices may bring between 200 and 400 percent improvement in productivity (Daly, 1979), others stipulate that if "there are numerous techniques that have been identified in the literature as yielding productivity gains up to about 25 percent, if they are applied seriously and conscientiously" (Jones, 1979, p.80), larger productivity gains are more difficult to obtain. Jones further states that to increase productivity by more than 75 percent, "only a few technologies remain as likely candidates, and even these may work only for special and selected cases" (Jones, 1979, p.81). Moreover, some studies showed that these programming practices are not widely used (Holton, 1977; Lehman, 1979). Hence, software productivity

remains a problem in search for a solution.

Going back to the queueing model, increasing the number of servers, namely the DP professionals, is another option which is available to DP in order to decrease user waiting time (given that the arrival rate and the average service time remain constant). However, firms have to face a shortage of programmers, and this trend will continue. Because of the steady decrease in hardware costs and the increase in manpower costs, the computerization of a larger number of applications becomes cost justifiable, and the demand for new application development increases dramatically ("Missing Computer Software," 1981; Martin, 1982). Nowadays, the increasing demand for new applications leads to an increase in the demand for DP professionals in general, and for programmers in particular. According to Martin, "if we assume no increase in programming productivity...in ten years' time the industry will need 93.1 times as many programmer as now....That suggests about 28 million programmers" (Martin, 1982, p.2). Martin further claims that the software firms attract and hire the best programmers, leaving the other firms in an even worse position.

The previous discussion of a DP department as a queueing system assumed that the arrival rate of the requests for application development and maintenance was constant. According to Martin, it is not so; he claims that according to an IBM survey "the number of documented applications in today's data processing centers is growing at 45% per year." Alloway's (1980b) findings, even if they do not provide an estimate of the growth rate of new applications, corroborate

Martin's assertion. As part of a larger research project on user needs (Alloway, 1980a; Alloway, 1980b; Alloway, Bullen, and Quillard, 1981), Alloway studied the composition of the applications backlog in six industrial firms. He defines two types of backlog: the known backlog and the invisible backlog. The known backlog consists of the backlog documented by DP. The invisible backlog is composed of applications users want but have not yet requested. Table 2.1 summarizes the results obtained.

These results indicate that the known backlog represents 68 percent of the applications already installed, while the invisible backlog represents 164 percent of the known backlog, that is, 112 percent of the already installed applications. According to Alloway, these results indicate that "systems development personnel are already flat-out dealing with this known backlog." Moreover, "the size of the invisible backlog implies that the known backlog will never get any shorter. No matter how fast DP can create new systems the users will keep the known backlog full." (Alloway, 1980b, p.14)

Another problem DP departments have to face lies in the "composition of the queue." That is, DP does not only respond to demands for the development of new applications but also has to maintain those already in place. Since systems are seldomly discarded, the proportion of DP resources devoted to maintenance tend to grow over the years. As soon as a system is implemented it becomes susceptible of requiring maintenance. According to Martin, maintenance "has become a nightmare in some large corporations" (Martin, 1982, p.84). Several

TABLE 2.1

INSTALLED SYSTEMS VS. BACKLOG

Total number of systems:

Installed:	}	277
		68%
Known backlog:		188
	}	164%
Invisible backlog:		309

estimates have been made on the amount of DP resources devoted to maintenance. Lientz, Swanson, and Tompkins (1978) report studies that found a level as high as 75 percent of resources devoted to maintenance. In their own study, Lientz et al. found that firms allocated an average of 50 percent of their DP annual manhours to maintenance. Twenty percent of the firms allocated as much as 85 percent.

Yet, a more interesting result for the present research pertains to the type of activities which constitute maintenance. According to Lientz et al. there exist three categories of maintenance activities. Those activities are defined as follows.

Corrective maintenance (performed in response to the assessment of failures); adaptive maintenance (performed in anticipation of change within the data processing environments); and perfective maintenance (performed to eliminate inefficiencies, enhance performance, or improve maintainability). (Lientz et al., 1978, p.466)

TABLE 2.2
BREAKDOWN OF MAINTENANCE ACTIVITIES

<u>CATEGORY</u>	<u>ACTIVITIES</u>	<u>RELATIVE FREQUENCY</u>
Corrective	Emergency fixes, routine debugging	17.4 %
Adaptive	Accommodation of changes to data inputs and files and to hardware and system software	18.2 %
Perfective	User enhancements, improved documentation, recoding for computational efficiency	60.3 %
Other		4.1 %

Table 2.2 outlines the breakdown of maintenance activities, in terms of the above categories, for the firms studied by Lientz et al. These results strongly suggest that, at least for the firms studied, more effort is put on maintenance activities which improve systems (60.3%) than on activities which merely rectify them (17.4%). Moreover, in the perfective category, enhancements requested by users accounted for two-third of the total. That is, a fairly large proportion of the maintenance load comes from changes in the user's environment, task, knowledge, or needs.

The Lientz et al. study provided another interesting result. Respondents to the survey (managers of systems and programming departments) were asked to rank possible problem areas in the context of maintenance. Users' requests for enhancements was the only problem ranked by the majority of respondents as being more than a minor

problem. While no details were given on what makes users' demands for enhancements a problem, the study showed that it was an important concern for DP managers.

DP departments also have to face the problem of user dissatisfaction with the applications, once they are developed. However, this discussion will be left for the next section of the chapter, that is, users' problems in the context of application development.

To sum up the previous discussion, DP managers have to face an effectively infinite demand for application development, with a limited quantity of manpower resources within their departments. Users are often dissatisfied because of the length of time they have to wait to see an application completed; during the period of application development, the users' needs may have changed. If so, users will request changes to be made to their applications, hence increasing the maintenance load. Given such a situation, many DP managers may be looking outside their departments to find elements of solution.

2.1.2 Users' Problems in the Context of Systems Development

The previous section mentions one problem users face in the context of systems development, that is, long waiting time before the applications they requested are completed. However, even when systems development is completed, users are often dissatisfied with the result. Over the years, much has been written on the DP professionals' failure

to provide users with the information systems they really need (Ackoff, 1967; Dearden, 1970; Keen and Scott Morton, 1978; Martin, 1982). In the literature, two main approaches are taken in attempting both to explain and avoid MIS failures. The first approach identifies the process of determination of information requirements as being the most important reason for which systems fail, while the second approach focuses on the implementation of the systems.

Determination of Information Requirements

DP professionals' inability to adequately identify users' information requirements is often identified as the most important reason explaining information systems failures (Taggart and Tharp, 1977; Bariff, 1977; Munro and Davis, 1977). As Munro and Davis state, "because the information provided is frequently the only interface which most users have with the system, information analysis is often the origin of perceived systems inadequacies....If the information provided by the MIS is perceived to be of low quality, the MIS is held in low esteem." (Munro and Davis, 1977, p.55)

Ackoff's (1967) argument on MIS failures puts emphasis on the errors made by systems designers in the determination of information requirements. A first "erroneous assumption" made by designers is that managers know what information they need. Ackoff argues that users do not know, and "playing safe" say they want everything. Moreover, the "MIS designer who has even less understanding of the relevant phenomenon than the manager, tries to provide even more than everything" (Ackoff, 1967, p.149). Instead of condensed and filtered

information, users are provided with masses of "irrelevant information." Designers also assume that managers need information about what other managers do. According to Ackoff doing so may negatively affect an organization's overall performance. To Ackoff, an important corrective to these errors is that the designer understands and models the decisions users make.

During the past decade, the issue of information requirements analysis (IRA), that is, the process of determining users information requirements, has received much attention. Davis (1974) identifies two approaches to IRA, namely the "data analysis" approach and the "decision analysis" approach. Data analysis, also called the bottom-up approach determines information requirements for a given application by analyzing the data which are currently used, as well as by asking the user to indicate what other data might be required. This approach has much similitude with the approach Ackoff argues that systems designers wrongfully take. On the other hand, the decision analysis (top-down) approach bears resemblance with the approach Ackoff suggests. Decision analysis attempts to elicit user requirements by analyzing and modeling the decision which is to be made. Research has been conducted in order to determine the most appropriate approaches (Kennedy and Mahapatra, 1975; Munro and Davis, 1977).

Several methodologies have been proposed in order to improve IRA. King and Cleland (1975) advocate a methodology wherein descriptive and normative system models are developed and are used as a basis for negotiating a "consensus system model." This methodology calls for

participation of the managers who will use the system. More recently, Grudnitski proposes a methodology which focuses on "making explicit, through measurement, the decision maker's cognitive structure of information sources" (Grudnitski, 1981, p.105). Bariff (1977) and Cooper and Swanson (1979) present and discuss other IRA methods. Several of these methods emphasize the importance of the user role in the context of IRA.

Implementation of Information Systems

Literature on implementation of information systems puts emphasis on the importance of behavioral aspects in the introduction of information systems (Argyris, 1971; Lucas, 1974; Ginzberg, 1975a,b). Argyris provides the following explanation to MIS failures.

The MIS per se is not the basic problem. The basic problem is that modern organizations, as we indicated at the outset, are designed with power centralized at the top, with specialization of tasks which results in many concealed dysfunctional components that are revealed by an MIS and that MIS implies a different design for organizations, one where competence is more important than power and collaboration and interdependence are more important than competition. This tends to create many fears and resistances on the part of individuals, groups, and intergroups. (Argyris, 1971, p.289)

Argyris argues that the approach taken by MIS specialists further contributes to create fears and resistances. While managers tend to make decisions based on "intuitive heuristic understanding," MIS specialists, with a mandate from top management, come with a highly rational approach to problems. Moreover, MIS specialists lack interpersonal competence. According to Argyris, interpersonal competence would contribute to create conditions with "minimal

defensiveness."

During the past decade, much effort has been devoted to studying MIS implementation, and two main approaches have been adopted. The first approach attempts to identify factors associated to successful MIS implementation. Lucas (1978), for instance, summarizes the findings of nine empirical studies he conducted in order to test a descriptive model of MIS implementation. From the results obtained, Lucas identifies quality of the system, favorable user attitude and perceptions, management support, as well as personal and situational factors as being key variables in the context of MIS implementation. Swanson (1974), found user involvement and attitudes to be directly related to system use. Other studies, such as Radnor, Rubinstein, and Tansik (1970), Bean, Neal, Radnor, et al. (1975), take a similar approach.

The second approach looks at systems implementation as a process of change. Studies based on the Lewin-Schein (Schein, 1969) model of change (Sorensen and Zand, 1975) or on the Kolb-Frohman (1970) expanded version (Ginzberg, 1975b) suggest that successful projects go through the three stages of the Lewin-Schein model. These results have implications for the systems designer who has to be more a manager of change than a technical specialist. One of Ginzberg's findings suggests that the role of agent of change is a difficult one to play. Ginzberg found a "low degree of perceptual congruence between users and designers where the users were dissatisfied with outcomes" (Ginzberg, 1975b, p.244). Moreover, the differences in perception "tend to be

largest at the process stages which appeared to be most critical for the success in that type of project" (p.245).

Summing Up Users' Problems

From the literature on both the determination of information requirements and on implementation, one common critical issue emerges, that is, the role played by the systems designer. Because an intermediary is needed between what the user wants and how the system is actually designed, communication problems are likely to emerge. In turn, such problems may lead to difficulties in adequately determining user information requirements. Furthermore, the differences in "culture" between systems designers and users often lead to implementation problems.

2.1.3 Emergence of UDA

The problems users and DP departments experience in the context of systems development call for solutions which address the programmer productivity problem while contributing to improve DP responsiveness to user needs. Very few solutions have the potential of achieving both. On the one hand, while methodologies which call for heavy user involvement in systems development have the potential of producing systems with which users are more satisfied, they do not address the productivity issue. On the other hand, modern programming practices do not tackle the problems of definition of information requirements and of implementation.

However, the past decade has witnessed several changes that contributed to making UDA become a feasible solution. One of these changes is the dramatic drop of hardware costs. Dertouzos and Moses (1979) claim that in the past 20 years, there has been an "incredible anti-inflationary trend in the computer field [that is] the reduction of the cost of the underlying components (hardware) that make up computers at the rate of about 30 percent per year, amounting to an overall cost reduction of a factor of one thousand for the same level of performance" (p.xi). Martin (1982) claims that in 1979, the curves of the cost of computer time and of the cost of people time crossed, making computer time cheaper than people.

Two important consequences result from the drop of hardware costs. First, the concerns of performance (with regard to programs) are not as serious for DP. Second, micro and mini computers have spread very rapidly, making UDA "easy and cheap." Seaman (1981) reports that in the United States, "almost 750,000 business persons are currently using microcomputers." Most of the time, the microcomputers come with application packages that are relatively easy to learn and to use.

The technical developments that Keen and Scott Morton (1978) suggest as having made the Decision Support Systems (DSS) concept feasible also apply to the context of UDA.

The most obvious of these is time-sharing. By the mid-1970s general purpose computers used operating systems that permitted nontechnical users to work at a terminal without needing specialized skills and knowledge. The machines have become increasingly

reliable, and "user-oriented" languages are widely available. More recently, versatile video display devices have replaced the slower and noisy teletype. Systems designers have also paid more attention to issues of human engineering, so that working directly with a computer is more comfortable. The development of such flexible, robust, and inexpensive tools has made possible "interactive" computer systems. (Keen and Scott Morton, 1978, p.12)

Of these developments, one has had a major role in making UDA feasible, that is, the availability of "user-oriented" languages. Some of these languages are suitable for queries only (e.g. QUERY-BY-EXAMPLE, from IBM), others combine query and report generation capabilities (e.g. EASYTRIEVE, from Pansophic), others are designed for model building purposes (e.g. IFPS, from Execucom), while others are query language, report generator, graphics generator, application generator, and high-level programming language all-in-one (e.g. FOCUS, from Information Builders).

Such technological changes, combined with the "increased computer literacy of users" (Rockart, 1976) have lead some users and DP departments to consider UDA as a feasible and promising opportunity.

2.1.4 Summing Up the Origin of UDA

Two phenomena contributed to create the need for UDA. Those phenomena are DP problems in the context of systems development and users' problems in the same context. Most DP problems, such as projects behind schedule, large backlogs of applications, heavy maintenance load, are related to the issue of programmer productivity.

One way of addressing the issue is to implement productivity improvement techniques. Another approach is to relieve DP professionals of some of their workload by redirecting some of the application development to users.

DP professionals' failure to develop systems that adequately respond to users' needs is the major problem users have to face in the context of systems development. Communication problems between systems designer and systems users may lead to inadequate determination of information requirements. The difference in background between user and designer may further lead to implementation problems.

Technological developments, including user-oriented languages and decrease in hardware costs have made UDA feasible. In addition to relieving DP from some of its workload, UDA has the advantage of eliminating the "middle man," and "users can create and modify their own applications as the need arises." (McLean, 1979, p.37)

2.2 UDA LITERATURE

Literature directly pertaining to UDA seems to naturally fall into one of the following categories: essays on UDA, reported experience of firms with UDA, and finally, research on UDA. Each of these categories is presented below.

McLean, McCracken, and Martin all draw a boundary around the domain of applications amenable to development by users: "Naturally, not all application development can be done by end users. There are always going to be the big, complex applications with interdependent end users that demand the skills of the trained DP expert" (McCracken, 1980, p.9). McLean labels such applications "corporate applications." In the case of those applications, the three authors put emphasis on the importance of heavy user involvement in the development process; they also suggest prototyping as being a means of improving the quality of the system that will be delivered to the user. Finally, they suggest that the use, by DP professionals, of high-level non-procedural languages in creating these applications should improve productivity.

Benefits of UDA

To Davis, UDA has three advantages:

1. Relieved shortage of systems development personnel.
2. Eliminates the problem of information requirements determination by information system personnel.
3. Transfers the information system implementation process to users.

(Davis, 1981, p.6)

In addition to the benefits identified by Davis, McLean suggests that UDA has the potential of eliminating "time consuming conferences," as well as reducing the portion of the maintenance load that is related to responding to users' requests for changes [1]. According to Martin, UDA does not only have the potential of decreasing DP workload and making application development a more flexible process. He also claims

2:2.1 Essays on UDA

McLean (1979), McCracken (1980), Davis(1981), and Martin (1982) present a view of UDA which is based on practical experience with DP and knowledge of the MIS field rather than on actual research on UDA. However, they go beyond mere reporting of facts, or sharing of people's experience with UDA. For those reasons, the word "essay" was chosen as a category label. It might be argued that Martin's book, Application Development without Programmers (Martin, 1982), is more than an essay. In fact, the book covers two categories of UDA literature included in this chapter: some parts are more of the essay type, while other sections are more of the reporting type. Each will be discussed in due course.

This section summarizes the views of the above authors with regard to four aspects of UDA. Each author's interpretation of the rationale for UDA is first presented. Second, potential benefits of UDA are outlined. Third, the requirements for facilitating the task of application development to users are discussed. Finally, dangers related to UDA are identified. Each author does not address every one of the above aspects of UDA: Davis, for instance, focuses on the dangers of UDA. Nevertheless, the above aspects encompass most of what has been said by the four authors.

Rationale for UDA

To McLean, UDA is the most appropriate response that a DP department can make to users' demands for more and better computer applications. Other means of responding to this increasing demand are to hire more programmers or improve programmers' productivity by using better techniques. The former is most difficult; DP departments are either unable to get approval for hiring more staff or they cannot find them. On the other hand, modern programming practices are valuable for applications where the information requirements are already well defined; yet, definition of information requirements is one of the most problematic phases of systems development, and this is mainly due to the fact that users have to "translate" their information requirements and communicate them to DP professionals. UDA has the virtue of eliminating this translation-communication process.

Martin and McCracken put forth similar arguments as a rationale for UDA: UDA is one way of addressing the programmer productivity problem. To McCracken, other means of solving the problem are to write reusable code, to move away from COBOL and PL/I, toward higher level languages, to build prototype systems, and to buy software packages. To Martin, the solution to the problem is to move towards application development without programmers. This solution includes (1) to have users develop their own applications, (2) to have systems analysts create applications (with appropriate tools) instead of writing specifications (which will be coded by programmers), and (3) to purchase application packages.

that UDA can bring "huge gains" in application development productivity. One instance of such gains is the case of one user department which created a complex application, using an application generator. The task took four months to two people. The user department claims that DP had estimated that if their own DP professionals were to develop the application they would require 20 people for 18 months, and cost \$1.5 million (Martin, 1982, p.30).

Requirements for Facilitating UDA

McLean states that for UDA to be beneficial, users have to be provided with proper tools, both hardware and software. Following the same line, McCracken proposes some characteristics the software tools should have. They should be user friendly, that is, easy to learn, with a natural syntax, and permitting to do much with only a few commands. They should be related to a data base management system, with a query language, a report generator, and the ability to ensure data validity and integrity.

Martin claims that DP should be the driving force toward UDA. DP management should encourage users to do application development, and make them perceive that UDA is beneficial to them personally (either because of time savings or because of better or more interesting work). DP should also provide good training to users and support them during application development. User-friendly tools, that can pass the "two day test" [2] should be made available to users. It is important that the various software tools be integrated: "A good query language should lead to a data base updating capability, report generation, and

graphics generation, using the same style of dialog and syntax" (Martin, 1982, p. 108). Martin stresses the importance of the concept of data base management, in the sense of a central responsibility for data coordination. According to Martin, data base management is necessary in order to assure consistency, accuracy, and security of data.

Dangers of UDA

Martin identifies two major dangers related to UDA, namely, duplication of data, and excessive growth in computer utilization. According to Martin, data duplication is likely to occur in situations where data base management does not exist. In such cases, files containing the same data fields may be created by several different users and given different structures. The result is likely to be systems between which communication will be difficult to establish. Data administration and data base management systems should exist in order to avoid this problem. Martin also raises warnings with regard to computer performance. He claims that powerful user languages allow "users to take actions that could burn up machine time." Another problem, related to capacity, may arise when, and if, there is excessive growth in utilization, and that the computer system cannot satisfy the demand. Martin suggests, as a solution to that problem, that there is a computer system devoted to users. Users "can do what they want with their own system and pay for it" (Martin, 1982, p.297). However, in the situation where a computer is to be devoted to UDA, Martin stresses the importance of maintaining user activities under the coordination of data administration. In such a case, data would be

made available through an "extractor program." Such a program provides the interface between the main computer system and the computer devoted to UDA by transferring data from the former to the latter.

While neither McCracken nor McLean address the issue of the dangers related to UDA, Davis' discussion emphasizes UDA dangers and risks. While the focus of Davis' paper is on user developed Decision Support Systems (DSS) in particular, the discussion is most relevant to UDA in general. According to Davis, by eliminating the role of the professional analyst as the system developer, UDA creates several risks. He further identifies five main causes of dangers, as well as their consequences.

The first cause of danger is the elimination of the separation of functions between user and analyst. As a consequence, the analyst is no longer there to play the role of independent reviewer of information requirements and to assure that organizational standards and practices, such as documentation, testing, controls, etc., are respected. The second source of danger is that there exist limits in user abilities to identify the correct and complete requirements for a DSS application. Davis argues that a good analyst has the knowledge, methods, and experience required to efficiently elicit information requirements. While some users may be as proficient in performing that task as good analysts, a large proportion of users will perform poorly. That is, because they have a poor model of requirements to bound the problem space and an inefficient search procedure within the problem space, they do not perform well. This poor performance of users may lead to

situations where there is too much analysis and inefficient search within the problem space, where the wrong problem is solved, or worse, where the wrong analysis is applied to the wrong model. The third cause of dangers is that users are unfamiliar with, or unwilling to apply the organization's quality assurance methods and techniques. In many cases, users may be reluctant to test and document their systems, to include audit trails in their applications, and to build in operating controls. Fourth, there exist risks related to the fact that user systems are likely to be unstable; that is, applications developed by users are likely to change as the user environment, needs, or knowledge change. Davis argues that because most organizational processes are dependent on stable systems, the use of applications developed by different users is not suitable for interdependent tasks. The last cause of danger mentioned by Davis is the risk related to the encouragement of private information systems, which leads to information hiding by individuals.

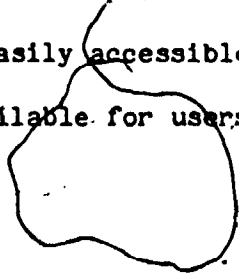
In order to minimize the risks discussed above, Davis suggests the implementation of controls: (1) review of applications by professional analysts, (2) formal policies requiring review and documentation, (3) user training in problem finding, quality assurance and controls, (4) automatic documentation procedures and quality assurance procedures.

2.2.2 Reported Experience of Firms with UDA

Most of the literature reporting firms' experience with UDA is related to the concept of Information Centre (IC), as initiated and promoted by IBM (Data Processor, 1979; IBM, 1981a). This concept has received much attention in the recent past. While the early writings dealing with the IC concept closely related it to IBM (Grindlay, 1980; McNurlin, 1981; Hoard, 1981), the tendency is now to see it as an independent concept rather than as an IBM product (Doubler and Cooper, 1982; Martin, 1982). Before summarizing the literature on firms' experience with UDA, this section first presents the IC concept.

Information Centre

The IC concept originated at IBM Canada, and its establishment was "a direct result of a major shift in DP strategy, by IBM Canada, resulting from the recognition that all the information requirements of a widely varying user community could not be satisfied by a central group of DP professionals" (Data Processor, 1979, p.6). The IC is both a location and a group of people. It is an organizational entity whose mission is to provide the user community with services that allow users to access and manipulate data. Such services include programming tools (user products, in IBM's terms), training, and ongoing support (Rosenberger, 1981a). The IC is also responsible for insuring security of data and sufficient computer capacity to fulfill user needs. According to IBM's philosophy, there should be a space set aside especially for the IC and this space should be easily accessible by the majority of users. Terminals should be made available for users who do



not have one of their own or one in their working area. IBM also puts emphasis on the statement that the IC staff should not write code for users (Grindlay, 1980), the main reason being that the IC should help users help themselves rather than becoming a parallel DP department. However, Martin (1982) has a different view of the role of the IC staff. He claims that the IC consultants should do some application development for users, using nonprocedural languages.

The main responsibilities of the IC staff are to assist users in the choice of the appropriate tool to use for a given application, to train users with the tools, to provide ongoing support during application development (debug assistance, "troubleshooting") and to facilitate access to data. The staff is also responsible for promoting the IC concept in the user community, evaluating new tools or services, assisting users in planning for their future use of DP resources, and being an interface between the user community and other departments within DP (IBM, 1981a; Rosenberger, 1981a). The latter responsibility includes interfacing with the operations group providing the computer resources, with the data base administration group, and with the systems development group (IBM, 1981a).

It is often stressed that interpersonal skills of the IC consultants are at least as important as, if not more important than, technical skills (Grindlay, 1980; Rosenberger, 1981a; Martin, 1982). This is due to the fact that the IC staff spend most of their time interacting with users, teaching them, trying to understand their problems, or explaining a solution to a problem. Moreover, since many

users are likely to know very little about computers and programming, the IC staff should be able to "make things look simple." It is also considered important that the IC staff have a good knowledge of the firm's organization and business activities (IBM, 1981a).

Martin and Rosenberger both suggest that the IC manager should report directly to the senior DP executive, as does the manager of conventional systems development. According to Rosenberger, "establishment of a separate organizational unit also projects an image of commitment to end user computing to both users and others in Data Processing" (Rosenberger, 1981a, p.653). However, the IC must keep close relationships with data administration and with systems development. Since data administration is responsible for making data available to the user community, the IC staff will be the intermediary between data administration and the users, and should, according to Martin, "participate in the logical design of data bases" (Martin, 1982, p.303). Close relationships between IC and systems development are necessary in order "to insure appropriate use of both alternatives" (Rosenberger, 1981a, p.653). Rosenberger further suggests that the IC should be a profit centre. The main advantage of charging users for services they receive is that, in this way, it is clear that it is the users' responsibility to justify usage of the resources. Making the IC a profit centre also is an incentive for the IC to provide a high level of service.

In summary, an IC is an organizational entity whose responsibility is to facilitate UDA, by providing users with tools and support, in an

environment controlled by DP.

Firms' Experience with Information Centre Concept

The content of this section is based on presentations made by IC or DP managers during GUIDE conferences (Holmes, 1980: Bank of Nova Scotia; Comper, 1980: Bank of Montreal; Magee, 1980: Consolidated Edison; Petersson, 1981: London Life). The information content of these sources may be different from what would be found in cases written by an independent observer. Moreover, it is often normative rather than descriptive. Yet, even if this literature has some shortcomings, it is representative of the conventional wisdom of IC managers.

In presenting the experience of firms with the IC concept this section focuses on the reasons for which ICs are created, their objectives, the implementation strategies taken, the products offered, corporate concerns and IC management concerns, and the potential benefits of the IC. Since each reference does not cover all the aspects, each will be cited when appropriate. Before going further, it should be noted that, while the four companies each introduced the IC concept, the name used for the support group varies (e.g. Timesharing Support at Consolidated Edison, Personal Computing at Bank of Montreal).

The reasons for introducing an IC vary among firms. As reported by Holmes, the Bank of Nova Scotia initially set up the IC in order to "offload from Systems Development the backlog of users requiring access

to already automated data" (p.1564). The Bank of Montreal had a slightly different motivation for introducing data retrieval facilities first and personal computing later [3]: "the data availability strategy at the Bank of Montreal is to make data increasingly more available to end users wherever they are located by providing them with the tools and skills to achieve self sufficiency" (Comper, 1980, p.1665). Consolidated Edison introduced the Timesharing Support group as a means of solving the problem of high outside timesharing expenditures. At London Life, "Data Processing has been directed to aggressively pursue Office Systems Technology and the Information Centre is seen as a key component of this strategy" (Peterson, 1981, p.739). Office Systems Technology is seen there as the most promising means of improving "people productivity and control the cost of doing business."

The IC mission is similar in all five companies, and can be summarized as follows: "to provide users without DP expertise with the tools and techniques they require to access the organization's computing power and machine-readable data" (Holmes, 1980, p.1560). More specific objectives in the case of the Bank of Nova Scotia were to improve users' productivity "through the exploitation of technology," to improve decision making capabilities and to offload Systems Development. For London Life, the goal of the IC is to "have a positive impact on the bottom line of the company's books" (Peterson, p.740). This goal is achieved by providing users with appropriate tools and support. At Consolidated Edison, in order to successfully "repatriate" outside timesharing users, the company's Timesharing

Support group is charged with "providing end-user computer services at a cost significantly lower than that incurred with outside services" (Magee, 1980, p.1352) while providing readily available support, user-oriented tools, and consistent response.

A common implementation strategy for ICs (and recommended by IBM, 1981a) is to start with a pilot test, involving a small number of users and systems (Comper, 1980; Magee, 1980; Petersson, 1981). Users who either had previous experience in dealing with UDA or who show interest in the concept are encouraged to become part of the pilot test. When, and if, the pilot test is successful, the IC is in a good position to get approval from top management to start its expansion.

A marketing approach is often taken with regard to the IC. Holmes, for instance, maintains that the IC ought to have a product strategy and direction. The product strategy adopted at the Bank of Nova Scotia was to provide relatively few products, since a "proliferation of products cuts down on the productivity and expertise of the Information Centre staff and therefore the quality of service" (Holmes, 1981, p.1564). Another aspect of the marketing approach is promotion. Both Holmes and Petersson relate the promotion efforts of the IC staff. In both cases, that is, at Bank of Nova Scotia and at London Life, making presentations to users was identified as being the most appropriate way of promoting the IC services. Bank of Nova Scotia not only made presentations to the user community but to the DP community as well. The reason for making presentations to DP people is that systems development people should be aware of the services

provided by the IC so as to redirect users to the IC whenever appropriate. The Bank of Nova Scotia also had a promotional UDA newsletter. In the two companies, promotion is used as a means of controlling growth: when the number of users grows too fast (for computer capacity available or IC support available), promotion activities are decreased. When it grows too slowly the IC increases its promotion activities. In the case of London Life, promotion was halted because the available IC staff resource was "used up."

Some companies offer a relatively small number of products while others offer a broad variety. For instance, Holmes reports that Bank of Nova Scotia's IC supports GIS, FPS, SAS, and PLANCODE; APL is available but not supported. London Life's IC provides APL/DI, ADRS-II, GDDM, SAS/Graph, IIPS, APL, GIS, and DCF, while the Bank of Montreal provides ADRS-II, APL/DI, DCF, and APL. Both ADRS-II and APL/DI are APL-based. (some of those software tools are described in Appendix G, along with other tools). With regard to the products available on the market, Holmes warns of "overstated ease-of-use of end-user products." She claims that "end-user products have not evolved sufficiently to be truly easy for non-data processing personnel to use" (p.1562). Moreover, the products are not integrated; for instance, "quite often the analytical products do not have data retrieval capabilities, so it is necessary for the user to learn two products for one application" (p.1564).

Areas of concern exist, due either to the fact that users develop applications themselves or because of the existence of a group to

support users. Holmes categorizes these concerns in two groups: corporate concerns and IC management concerns. Corporate concerns are related to data security, the risk of management making decisions based on erroneous reports (because users may make mistakes), DP resources wastage (CPU time, IC staff time), and user time wastage (Do users develop the applications they should?). In the same category, Comper also mentions data security, correctness of the information provided by the applications, and misunderstanding or misinterpretation of the definition of data being accessed. Comper also mentions the fact that "erroneous or careless processing of large amounts of data could result in excessive charges for processing" (p.1658).

The second category of concerns are IC management concerns. Apart from the overstated ease of use of tools, Holmes mentions a number of other concerns the IC manager may have with the IC. The first one is the re-justification of the IC existence to top management, that is, to demonstrate that DP resources are used in a way profitable to the firm. According to Holmes, charging users for the services they receive "removes the demand on an IC manager for the ongoing justification of the Centre" (p.1560). Other concerns are insuring the availability of sufficient resources of users, insuring security, and negotiating "service contracts" with operations so that users are at least as well served as if they were using a service bureau. Finally, the IC manager has to resist pressures from some end users to have the IC staff write code. Petersson relates London Life's experience with regard to that issue, where, by a management directive, a special group was put in place to write code for users. Petersson states: "After about 6

months of experience we have found this is not working well at all....I would not recommend this sort of 'project support' to others and hope to be able to eliminate it at London Life" (p.741). The main reason for which writing code for users is not advisable is that users do not take charge of the system once it is developed. Consequently, the IC has to maintain these applications and sees itself in the same situation as a DP department. To Petersson, it is very important that the IC keeps close track of usage and benefits in order to plan for future capacity requirements and to justify the IC existence to top management.

According to Comper, the main benefit Bank of Montreal is obtaining from Personal Computing is to allow users to develop applications in a "timely and cost effective manner, using minimal data processing personnel resources." To London Life, the IC services have the potential of increasing users' productivity and improving decision making. Consolidated Edison estimated that in the "best case" in-house Timesharing Support could reduce annual outside timesharing expenses by \$4 million, bringing them to less than \$1 million per year. According to Holmes, the main potential benefit of the IC concept is to reduce the DP maintenance load, so that DP can be more responsive to users' needs with regard to new systems development and the automation of more data.

Rosenberger (1981b) presented a paper entitled "The Productivity Impact of an Information Centre on Application Development." Rosenberger's argument is based on what has been reported by ICs

"around the world." Since Rosenberger's discussion is directly related to firms' experiences with the IC concept, it is included in the present section.

According to Rosenberger, the IC concept does not only have the potential of increasing users' productivity; it also has the potential of increasing productivity in DP departments. He identifies 10 "productivity factors:"

1. Leveraging of the DP staff resource
 2. "Skimming" the application backlog
 3. Maintenance reduction
 4. Development reduction
 5. Headcount avoidance
 6. Improved Return-on-Investment basis
 7. Off-loading reports to the user
 8. Lowering the justification threshold
 9. User prototyping as a design aid
 10. Shifting the development "mix"
- (Rosenberger, 1981b, p.919)

Leveraging of the DP Staff Resource. According to Rosenberger, leverage is reached when DP moves from traditional development, where the user makes a request and DP develops the application, to a situation where users develop applications themselves. He gives the example of IBM Canada, where, after 7 years of experience with the IC concept, 25 percent of the computing resources used for administrative applications are used by IC users. To provide the support for this 25 percent, there are 11 IC consultants and a total of 220 people working in application development. As put by Rosenberger, 5 percent of the staff supports 25 percent of the application load.

"Skimming" the Application Backlog. Rosenberger reports that one company claims an actual reduction in the application backlog by 50 percent after 9 months of IC operations, while another company reports that program requests decreased by 30 percent as a result of the use of the IBM user product Query-By-Example. However, according to Rosenberger, in most firms, users will tend to develop those applications that are part of the "invisible" backlog, as defined by Alloway (1980b) [4]. As a consequence, the IC "skims off the top" of the backlog those small, ad hoc, ill defined projects, and sets DP free to focus on large development projects.

Maintenance Reduction. Rosenberger claims that reduction of the maintenance load is "the most direct and obvious effect of the Information Centre within DP" (p.921). He reports IBM Canada's experience, where from 1973 to 1980, the amount of DP resources devoted to maintenance went from 70 percent to 40 percent. Other firms claim still more dramatic reductions: one company reports a reduction of maintenance from 70 percent to 40 percent in one year.

Development Reduction. The IC concept also has the potential of saving development effort, if the firm adopts what Rosenberger calls Shared Application Development, wherein DP "builds that portion of the application that processes transactions, produces required deliverables, and maintains the corporate data base while the user designs, creates, and maintains the reports required to satisfy business needs" (p.655). One company where this approach was taken claims to save up to 30 percent of the development effort.

Headcount Avoidance. Rosenberger claims that UDA can lead to a slower growth in the number of employees in DP departments. He reports IBM Canada's experience where the computer workload grew of 25 percent while the headcount stayed the same within DP, during a seven year period.

Improved Return on Investment Basis. Rosenberger reports the results of a user survey conducted at IBM Canada in 1978. One purpose of the survey was to determine an overall ROI for the applications developed by users, and to compare it to the ROI of 37% used by DP to justify traditional projects. The ROI calculations were based on headcount savings and tangible benefits only. Intangible benefits were excluded from the calculations. Costs included user headcount and processing costs (CPU time plus overhead). A value of 100% was obtained for the ROI. From this result, Rosenberger states that "for the appropriate type of applications, then, the Information Centre appears to offer an attractive ROI opportunity." (Rosenberger, 1981b, p.922)

Off-loading Reports to the Users. Rosenberger claims that important gains can result from providing users with tools for querying and ad hoc reporting. He gives the example of a user department at IBM Canada which reported a cost ratio of 15.7 to 1 in favor of UDA (i.e. for a given number of queries and ad hoc reports, development by DP would have been 15.7 times more costly than development by users). Rosenberger argues that the high value of the ratio is first explained by the fact that users required less time to develop the applications. While DP would have used PL/I, users were provided with GIS which is a

more efficient tool. Second, while DP's monthly rate for development was \$4,000, users estimated their monthly development cost at \$1,400. Rosenberger adds that report generation by users also cuts the "hidden cost of bureaucracy," since users no longer have to go through long procedures to submit a request, have it approved, and so on.

Lowering the Justification Threshold. This benefit is due to the fact that users have lower development costs than DP. Consequently, some applications, which development would not have been justifiable on the basis of DP development costs, become justifiable if they are to be user-developed.

User Prototyping as a Design Aid. Rosenberger claims that products available to users through the IC make prototyping possible, both to demonstrate the feasibility and to validate the design of Decision Support Systems and Personal Computing systems.

Shifting the Development Mix. According to Rosenberger, "the IC concept is a means by which DP can provide the user community with a service that users wanted but that DP did not offer. From Alloway's (1980b) study, it appears that users' needs are shifting from traditional operational control systems (which Alloway calls Monitor and Exception) toward "management support system" (which he identifies as Inquiry and Analysis systems). In order to satisfy these needs Rosenberger argues that DP has to provide users with a "pluralistic service strategy." Otherwise, the users "will flock outside timesharing or install minis in growing numbers" (p.925): This would have the consequence of

reducing DP's control of computing and might also lead to higher costs for the corporation as a whole.

Other UDA Experience

This section presents experiences with UDA that have taken place outside the context of an IBM-like IC. These experiences are reported by McNurlin (1981), Seaman (1981), and Martin (1982). McNurlin presents two short cases describing companies where DP made tools available to users. Seaman describes the experience of half a dozen managers who use microcomputers, while Martin presents a case where a user department created a complex application without any help from DP.

Mouser Electronics. The first example presented by McNurlin is a small wholesale distributor of electronic components, where the DP staff is composed of the Vice-president Operations and Data Processing and one programmer. At the time of investigating alternatives for upgrading the computer configuration, the Vice-president decided to acquire a PRIME 400 with the "Information" operating system. User friendliness and ease of expansion of the computer system were two important characteristics that the chosen system had to have. Ease of use for non-DP users was very important, since the company had only one programmer. The Vice-president, after receiving training himself, trained the company managers in the use of the "English-like non-procedural language" available on the system, as well as some application packages. Once trained, the managers trained their support personnel. Less than two years after the installation there were more than 800 programs written, many of them by users. The main benefit the

company has received from it is a "better control of their business because of increased use of the computer." Moreover, the programs users wrote themselves were available much more quickly than if they had had to wait for the programmer to write them.

Lincoln National Life. The second case relates the experience of Lincoln National Life, an insurance company that employs 3,500 people and has assets of \$4.3 billion. At Lincoln National Life, UDA is part of a program for "distributed processing and office automation." With regard to UDA in particular, the company decided to offer users five facilities through a single menu-driven front-end: (1) Electronic mail (2) Word processing (3) Electronic filing (4) Records management, and (5) On-line tutorial. The system operates on PRIME departmental computers, and makes use of INFO, "a file management system and programming language." One manager and three staff members are responsible for training and supporting users. Acceptance from users was very good: DP cannot install systems fast enough to satisfy the demand. In general, the programs users write are short and use small files.

Microcomputers Invade the Executive Suite. Such is the title of Seaman's (1981) article, in which he reports the experience of managers who have acquired microcomputers. While using a microcomputer does not automatically imply writing code or developing applications, most of the managers interviewed by Seaman had created at least some of the programs they used. According to Seaman, almost 750,000 businesspeople were using microcomputers, at the time the article was published. He

argues that several of the businesspeople who acquire a microcomputer did so because they were dissatisfied with the services provided by the DP department in their organization. Moreover, many use their own money to purchase a desk-top computer, while others simply "bury" the cost so that it does not become visible to DP. Given the relatively low price of microcomputers, their cost is generally not hard to hide. Seaman reports the experience of half a dozen of managers with microcomputers.

Some of these users had had applications developed for their microcomputer by DP people in their firm, some used pre-written application packages, and most also developed some applications themselves. They use their microcomputers mainly for analysis activities, to evaluate different scenarios when making decisions, and so on. While the majority of these managers have a technical background (such as engineering), one of them is a lawyer, with no technical background. Even so, he developed his own applications on a Radio Shack computer, which he uses in preparing various options for estate planning for his customers. According to Seaman the microcomputers are used mainly for personal computing. One of the managers interviewed claims that calculations that took him several hours using pencil and paper and a hand calculator now only take one minute, once the data has been entered. Another manager stated that using a microcomputer let him "tailor the information to precisely the format I need," rather than having to accept the format provided by DP. This manager also claimed that his decision making was improved, since he has eliminated "superfluous and redundant information which just

confuses the issue." He inputs himself data from reports provided by DP into his microcomputer. He hopes that one day he will be able to have access to corporate files through his desk-top computer.

Poor documentation on how to operate the computer has been a problem for one user, while another one argued that it is difficult to select the appropriate software packages to use, since there are so many of them on the market.

A Factory Production Control System. Martin (1982) reports the experience of a user department that developed a complex production control system without receiving any assistance from DP. Users in that department (dispatchers, foremen, expeditors, and other staff) had asked the company's DP department to develop an on-line production system, in order to rejuvenate the current pencil, paper, and "huge computer listings" system. DP indicated that it would take them a few years to complete such a system. The department then decided to rent a computer and to develop the system internally. A UNIVAC computer was rented, and UNIVAC's MAPPER application development software system was installed. The DP department didn't think that this experience would be successful. Nonetheless, users in the department proceeded to develop a sophisticated production control system, that included work-in-progress, tracking of orders through the factory, control of physical inventories, and so on. The system saved much time to department's staff, with regard to report preparation. Moreover, Martin reported that users could have much better control of the "movement of work through the factory," and that the system assisted in

speeding up orders, because of the accuracy and completeness of the information provided.

The manufacturing department users were so enthusiastic about their success that they started selling the idea to other departments. Martin reported that the number of users grew from four to several hundred in many different areas such as stockrooms, purchasing, and marketing.

According to Martin, the DP department remained skeptical for some time. However, after three years of successful use of MAPPER, "DP wanted to take it over so that they also could create applications faster."

Summing Up Firms' Experience with UDA

The examples provided in the previous two sections were reported either by practitioners who have actually experienced UDA (Holmes, 1980; Comper, 1980; Magee, 1980; Petersson, 1981), or by observers (Rosenberger, 1981b; McNurlin, 1981; Seaman, 1981; Martin, 1982). Each of these "reporters" has his or her own opinion of what is important in the context of UDA. The purpose of this section is to extract, from these several points of views, the main elements of "conventional wisdom" on UDA.

The first element of UDA conventional wisdom is that DP should encourage and provide some coordination to UDA. In the examples previously presented, the four companies with an IC, as well as Mouser

Electronics and Lincoln National Life, are cases where UDA has taken place under DP's "supervision," while the Factory Production Control system and the users of microcomputers are cases where UDA has taken place without coordination with DP. McNurlin argues that the latter is likely to have negative effects, such as "fragmented departmental data, poorly written user programs, and hardware that cannot be connected to a company's data network" (McNurlin, 1981, p.5). Rosenberger (1981b) gives the following warning: if DP does not provide users with UDA facilities, they will make more use of service bureaus. The result is that DP will lose some control of corporate computing and that company costs will be increased. Martin is also in favor of DP encouraging UDA.

The second element of UDA conventional wisdom pertains to user friendliness of the tools available to users. McNurlin argues that "the end user language has a lot to do with the users' acceptance of the system." Most of the examples given in the previous sections included the element of user friendliness of the tools.

The third element pertains to the support provided by DP. The IC's main role is to provide users with training and assistance. Even in the cases of Mouser Electronics and Lincoln National Life, where there is no IC but where UDA was taking place under the supervision of DP, users were supported by DP. In the case of the Factory Production System users did not receive support from DP, and neither did most of the managers who had their own microcomputer in Seaman's article. However, in the Factory Production System case, it seems that MAPPER is

so easy to use that no support was required. In the case of the microcomputer users, it was sometimes the technical background of the user which could explain that assistance from DP did not seem to be a major requirement. From that, one could hypothesize that DP assistance is less important when the tools are very easy to use or when users have a good background with computer usage.

Data administration, data base management systems, data base query languages are phrases that are part of almost all the examples given in the previous sections. Consequently, the fourth element of the UDA conventional wisdom is data administration. McNurlin stresses the importance of the data administration function which should have control over corporate data definitions, as well as control over data security. As seen in section 2.2.1, Martin also consider data administration as an important element of UDA. The experience of firms with the IC concept suggests that data availability and data security are important.

The fifth element is related to control. Control pertains to various aspects of UDA, such as control of outside computing resources expenditures (Magee, 1980), control of the growth of the number of users (Holmes, 1980; Petersson, 1981), control of access to data (Comper, 1980; Holmes, 1980), quality control of the applications developed by users (Comper, 1980; Holmes, 1980), and so on. The appropriateness of charging users is discussed. McNurlin reports that some companies, in order to encourage UDA, do not charge users at all, while others don't charge them for the first year. Holmes argues that

not charging users is "a good way to get an Information Centre started," yet, "a chargeout system removes the demands on an IC manager for the on-going justification of the Centre" (p.1560). Rosenberger (1981a) advises DP departments to charge users, in a way that is easily understandable by user management.

The sixth element of the conventional wisdom on UDA is the benefits dimension of UDA. From the firms' experience reported here, it appears that improvement of user productivity is a common element, and improved decision making was also mentioned quite often. For DP, Rosenberger's (1981b) list of 10 potential benefits covers more than all of the other benefits mentioned. Reduction of the maintenance load is a potential benefit which is often mentioned, as well as the reduction of the backlog.

2.2.3 Research on UDA

Up to now, very little research has been done on UDA. Apart from the present study, there is only one other ongoing research effort in this area: the research undertaken by the Centre For Information Systems Research (CISR) at M.I.T. CISR's research project on the management of end user computing started in early 1980 and has been ongoing since then. The approach they have taken has been to conduct field interviews in firms where "significant end user computing is being done" (Rockart and Flannery, 1981). To date, a study in one of the seven companies included in the research project has been completed (Stork and Bland, 1980) and the findings published (Rockart and

Flannery, 1981) [5]. The main findings of their study are presented below. [6]

APEX Study

The CISR study was conducted at APEX (company name disguised), a manufacturer of information processing products, with multi-billion dollar sales. At APEX, computing hardware is centralized and systems development decentralized. TIMS, the Timesharing Information Management Services, was put into place approximately four years before the APEX study was conducted. The main reason for creating TIMS was to "repatriate" outside timesharing usage. Before the creation of TIMS, timesharing was available in-house. However, it did not offer services comparable to those offered by service bureaus. This gave rise to the growth of outside timesharing usage. TIMS was given the mandate of providing users with services competitive with outside service bureaus.

According to Rockart and Flannery, management at APEX encouraged the study of end user computing on their premises for the following three reasons:

(1) the need to understand the factors underlying the surprisingly high rate of growth of, and therefore the company's increasing (and perceived as somewhat out of control) financial commitment to, end user computing; (2) the belief that significant risks were involved in having end users develop some systems; and (3) the desire to understand how end user computing should be managed. (Rockart and Flannery, 1981, p.354)

The main issue with regard to management of end user computing

pertained to how the user support organization should be structured so as to both serve users and anticipate their needs. The objective of the CISR study was to better understand end user computing at APEX by "determining the factors that are fueling the dramatic growth in end user programming and through exploring some of the key issues surrounding the management of the activity" (Stork and Bland, 1980, p.10). Rockart and Flannery report the study findings under four main headings: (1) reasons for timesharing growth, (2) end user profile, (3) application profile, and (4) key managerial issues.

Reasons for Timesharing Growth. The study identified five main factors that contributed to explaining the growth of timesharing usage. The first two factors are directly related to the main end user programming tool used at APEX, that is, APL. First, using APL makes it possible to develop applications rapidly, and second, to develop prototypes easily. The third factor is the "ease of access" to the timesharing system and rapid program turnaround. The fourth factor, and according to Rockart and Flannery, possibly the principal reason for the growth of timesharing use, is the "non-responsiveness of the traditional IS [Information Systems] community." At APEX, in addition to providing assistance to users who developed their own applications, the timesharing support group also performed application development for users. Users found that the timesharing support group was more helpful and cooperative than was the DP group. The fifth factor is the "expandable supply of programmers within the user organization." While the head count for professionals in DP may be restricted, it is not so for the user community. User departments can hire "financial analysts"

who will actually be full-time programmers. They can also train employees within their department to do application development work.

End User Profile: Users of timesharing at APEX were found to fall into four categories; (1) "Unsophisticated end users" represented approximately 70 to 80 percent of the timesharing users. These users do not perform any application development but run applications developed by other users or by the timesharing support group. (2) "Sophisticated end users" develop applications for their own use, and they sometimes help other users. They represent approximately 10 percent of the timesharing user population. (3) "Local functional support group" are users who have become APL experts and have become "informal centers of design and programming expertise within their functional areas." Yet, even if they spend a large proportion of their time programming (up to 80 percent) these individuals do not perceive themselves as DP professionals. This group of users represents between 5 to 10 percent of the user population. Finally, (4) "the professional support staff" is located in the Systems Support group for each division of the company. This category of users have, as their formal responsibility, to develop and maintain timesharing applications for end users. They represent 5 percent of the timesharing user community.

Application Profile. The study identified six broad categories of applications developed by the user community. (1) APL "prototypes" were developed, yet never replaced by a COBOL application. "Prototype" was often used "to mask a quick and dirty system that ends up lasting years" (Rockart and Flannery, p.357). (2) APL routinized batch

-systems, (3) general purpose models and statistical routines, as well as (4) "report reformatters," which merely reformatted reports originally produced by DP, were identified. (5) Inquiry and status reporting applications, which are "characterized by a large specialized data base which is accessed through a menu of subroutines heavily oriented to the provision of canned reports" (Rockart and Flannery, p.358) were also uncovered. The final category of applications is constituted by (6) text-oriented applications, such as electronic mail, text editing and processing.

Key Managerial Issues at APEX. Rockart and Flannery identify seven major findings that they describe as significant, company-specific issues. These seven findings are listed below:

1. Many applications vital to operational decisions have been programmed by end users. In general these applications are poorly documented and supported....
2. Informal support groups, who spend all their time performing systems development and programming functions, but who do not perceive themselves as IS professionals, pose several managerial problems... [since] they do not consider themselves bound by any of the normal constraints on computer professionals....
3. Users are willing to pay significant hardware running cost premiums to get systems up and running quickly under their control....
4. Locating data, and coordinating its collection, are problems for both new and experienced users....
5. Little attention is paid to transferability of applications, [i.e. sharing applications with other users, departments] resulting in considerable duplication of development efforts....

6. There is a mismatch between the types of applications being programmed and the end user languages available....

7. The current approach to the management of end user computing is unclear to both users and IS management....

(Rockart and Flannery, 1981, pp.358-361)

A Stage Theory for UDA

From the APEX study and from the information already gathered in the six other firms that are part of the larger CISR research project, Rockart and Flannery suggest that Nolan's (Gibson and Nolan, 1974; Nolan, 1974a) stage theory applies to the management of end user programming. [7]

Nolan's (1974a) stage theory identifies four stages of growth: Initiation, Contagion, Control, and Integration. Stage I, Initiation occurs when computing is first introduced into a firm. Computer usage is dedicated to a particular class of applications, generally of accounting type, such as payroll, billing, etc. Control of the DP resources is generally given to that part of the firm that provided the justification, very often the company's controller. One issue of that stage is computer underutilization. "The forces for rapidly developing 'production' applications create an environment which discourages planning, cost control, and quality assurance" (Nolan, 1974a, p.54). The introduction of computer technology is often disruptive, and sometimes alienates some people. During that phase, senior management acquires experience with managing DP, and becomes committed to computing. "This managerial commitment...marks the transition" to the

next stage, Contagion.

One common manifestation of this transition is the promotion of DP management in the organization's hierarchy, as well as a broadening of DP management's responsibility. In Stage 2, DP tries to promote its services and to get more "clients." Planning, project control, and budgetary control are generally non-existent. The "mot d'ordre" is to use up the excess capacity. Yet, the more effective these inducements are, the faster DP will reach a bottleneck, where the computer is saturated but users keep on requesting larger and more complex systems. Additional specialized and expensive DP professionals have to be hired. "The budget for the computer organization increases exponentially and quickly soars to a point of crisis concern for management" (Nolan, 1974a, p.56).

The third stage, Control, "is often an overreaction to the strong forces toward centralization." Top management becomes more involved, often by setting up a senior management steering committee, to evaluate what is being done and to set priorities for the future. Standards for systems development, programming, documentation, operations, and project management are established. DP starts charging users for services provided. Serious efforts are made to slow down the exponential growth of DP expenditures.

In the fourth stage, Integration, maturity is reached. Controls are refined, and some of the more arbitrary ones are removed. DP planning becomes an integrated part of corporate budgeting and planning

process. DP management becomes more user-oriented, putting more emphasis on what users need than on what computer technology can do. Chargeout systems that had been implemented during the control stage are sometimes modified to partial chargeout or simply abandoned. "The most dominant characteristic of stage 4 is a rethinking of the role of the computer resource with respect to organizational goal achievement" (p.57).

Rockart and Flannery argue that the results they obtained in their research on end user computing follows a "learning curve" much similar to the one described by Nolan's four-stage model. They claim that nowadays firms still are in the Initiation stage. These firms are just starting to make programming tools available to users, and few users are involved in end user computing. Those who are involved generally are "expert staff people" who do not require much assistance with the use of the tools. The user community uses a small amount of the firm's computing hardware resource (between zero and ten percent of the computing power).

Rockart and Flannery suggest that a "very clear set" of firms are now in the second stage, Contagion. These firms have been providing users with timesharing and other facilities for five to ten years. Between twenty and forty percent of the computing power in these firms is used by end users, and the growth rate is accelerating.

Rockart and Flannery do not report any company as being in stages 3 or 4. They stress the importance of understanding the key management

issues of end user computing in order to avoid the "debilitating and often devastating" problems that occur when the transition to the third stage is not properly managed.

2.3 OTHER AREAS RELEVANT TO UDA

The Information Centre is the direct result of a major shift in DP strategy by IBM Canada, resulting from the recognition that all the information requirements of a widely varying user community could not be satisfied adequately by a central group of DP professionals. (Data Processor, 1979, p.6)

The Information Center provides the opportunity to distribute computing capability to the end user with standardized hardware and software and centralized data management control. (Bradish, 1981, p.656)

The change in methodologies that this book describes encounter great emotional resistance in many installationsHigher DP executives can be alarmed by it also and sometimes envision their empire dwindling if end-user departments learn to create their own applications. (Martin, 1982, p.76)

These quotations suggest the following. First, how the firm deals with the concept of UDA is an intrinsic component of DP strategy. As such, it should be examined in the larger context of centralization, decentralization, and distribution of DP. Second, the introduction of UDA, either by DP or by users themselves, is an important change for an organization's members, both in the user community and in DP. These two aspects of UDA suggest two areas of literature relevant to the concept of UDA: the literature on centralization, decentralization (and distribution) of DP and the literature on organization change. The previously reviewed literature on UDA also suggests two other

relevant areas, namely, data administration and human-computer interactions.

This section gives an overview of the literature in these four areas, and indicates their relevance to the concept of UDA.

2.3.1 Centralization, Decentralization, or Distribution of DP

Centralization versus decentralization of DP is an issue that has generated much controversy over the last 15 years. In the late 70's, a new concept emerged within the computer industry: Distributed Data Processing (DDP). This concept is often said to have the potential of resolving the major problems of both centralization and decentralization. This section presents the major arguments of the centralization vs. decentralization controversy and indicates how DDP fits in it. The relevance of this issue for UDA is then discussed.

Centralize, Decentralize, Distribute?

Before presenting the classical arguments of the centralization vs. decentralization controversy, some definitions are proposed:

In a centralized system, all the computing resources of the organization are in a central location....In such a system, the central staff not only provides all technical services, but also plays a dominant role in setting development priorities and allocating computing resources among competing demands. (Sircar, 1979, p.79)

In a completely decentralized system there is no co-ordination or direct communication between multiple stand-alone computers. (Sircar, 1979, p.79)

Distributed processing is the deployment of computerized data processing functions where they can be performed most effectively, at costs usually less than those of other options, through the electronic interconnection of computers and terminals, arranged in a network adapted to the user's characteristics. (Kaufman, 1978, p.9)

The classical arguments of the centralization vs. decentralization controversy are well covered by Glaser (1970), Norton (1972, 1974), and Fried (1976). These arguments are presented here using Norton's classification: Systems Operations, Systems Development, and Systems Management.

Systems Operations. In the context of the debate on centralization vs. decentralization of systems operations, Grosch's Law has long been a major argument in favor of centralization. The law states that performance of a computer increases as the square of its cost (see Fried, 1976, for a discussion of the "law"). Large computers are also said to have the advantage of being able to process sophisticated applications requiring capacity that smaller ones do not have. Large installations are apt to attract and keep highly qualified technical personnel. On a corporate basis, one large installation is said to require less personnel to operate and manage than several smaller installations. The total manpower cost would then be lower, and so would be the total "computer environment" cost.

On the side of decentralization, it is argued that large installations are often underutilized, and that consequently the

effects of Grosch's Law are diminished. Moreover, the recent dramatic decrease in hardware costs further contributes to diminish the effects of the law. It is also argued that, on the whole, several single purpose minicomputers would be less expensive than a larger one. Decentralization reduces telecommunications costs and gives the users better performance (better turnaround, faster response) because they do not have to share the computer capacity with the entire corporation. Failure of a central computer would affect all users, while in a decentralized environment, fewer users are affected by the failure of a single CPU.

Systems Development. Regarding the analysis phase of development, centralists and decentralists seem to agree that decentralized analysis is the best alternative. Local analysts are more knowledgeable of local users' needs. Regarding design and programming, some controversy exists. Decentralists argue that if the whole system development activity is decentralized, (1) the systems will better meet local users' needs, (2) the systems will be available more quickly, (3) the systems developers will feel more as part of the firm than as part of their "trade," and (4) the users will better understand the possibilities and limitations of DP.

On the other hand, centralization of design and programming would allow the utilization of common data, common standards, and the use of systems development control techniques. Once again, a large central DP organization would attract highly qualified personnel who could develop more sophisticated systems.

Systems Management. It is the control aspect of systems management which is emphasized by both the centralists and decentralists. Centralized installations are easier to control because standards are common to the whole corporation. Decentralization makes costs more difficult to control because they sometimes may be recorded in other organizational components (manufacturing or accounting for instance). The main argument of decentralists is that in a centralized environment division managers will not have incentives to control DP costs. On the other hand, if systems management is decentralized, division managers will feel responsible for controlling the use of the computing resources, and will have direct insight into all the elements of DP costs.

Overall, from the vantage point of DP management, centralization would appear as being the best alternative. Systems can be provided in a very efficient way, supported by highly qualified personnel. The arguments presented above also indicate that control is an easier task in a centralized environment. On the other hand, for the end user, it appears that decentralization would bring more advantages.

However, some authors (Fried, 1976; Buchanan and Linowes, 1980a) argue that the organizational mode of DP "should be made congruent with the company's management style, organizational design, corporate objective, and user's need" (Fried, 1976, p.18). Buchanan and Linowes claim that the information systems of a firm should match the firm's strategy and structure. In order to achieve such a fit the authors argue that "careful attention must go into planning the arrangement of

the data processing resources that develop and operate [those] information systems" (Buchanan and Linowes, 1980a).

DDP. The centralization vs. decentralization issue has been partially resolved by developments in the computer industry. The most important has been the drop in hardware costs: the economies of scales due to a large computer system are not as large as they were, and in fact may have completely disappeared. This implies that for the price of a large installation a firm may have several minicomputers at different locations. This development alone is not sufficient to resolve all the issues related to centralization vs. decentralization of DP. Some argue that it makes the whole question even more difficult to resolve. "This [computer proliferation] can lead to duplication, lack of standards and generally poor control." (Withington, 1980, p.152)

However, technological developments such as advances in the areas of telecommunications and distributed data base management systems, have made feasible the linkage of these several minicomputers. This interconnection of computers is an important aspect of the concept of distribution of DP.

DDP makes it possible for certain applications to be developed and processed in locations that are remote from corporate DP, while other DP functions remain under the corporate DP responsibility [8]. A major advantage of distributed processing is that it has the ability to create happier relationships with users. As stated by Kaufman (1978)

DDP creates an environment that is responsive "to local and immediate needs, viewed emotionally, as OURS rather than THEIRS" while allowing overall control to reside in corporate DP's hands.

Positioning UDA with Regard to Centralization vs. Decentralization

Given the definition of DDP provided at the beginning of this section, UDA would be a way of distributing DP only in very special instances. Such instances would be cases where end users develop their own applications and run them on computers that are located at remote sites and linked to a corporate computer. However, the concept of UDA encompasses more than that, and so, to many people, does DDP. Consider the definition of DDP provided by Buchanan and Linowes.

To describe the data processing in an organization as "distributed" is to say that authority over one of the areas of responsibility...has been vertically or horizontally decentralized. (Buchanan and Linowes, 1980a, p.146)

The areas of responsibility to which Buchanan and Linowes refer are: (1) development, which includes data base administration, applications programming, systems analysis, systems documentation, and user training, (2) operations, including hardware operations, telecommunications, systems programming, and application system maintenance, and (3) control, which includes providing security, setting priorities, standardizing tasks, accessing data, scheduling tasks, planning personnel, budgeting, and evaluating products (p.146).

Rockart (1976) takes a similar approach in analyzing the

centralization vs. decentralization issue. He argues that the decision whether to centralize or decentralize DP should not be considered in a monolithic fashion. He suggests that each of systems operations, systems development, and systems management can be "centralized or decentralized almost independently from the others."

For a DP department, the decision to encourage UDA or not is similar to the decision whether to distribute DP, and if so, what to distribute. If DP decides not to encourage UDA, some users may acquire microcomputers or acquire computer time from service bureaus. In such a case, with regard to the applications users develop themselves, DP will in effect be decentralized. And the consequences of this may very well be those suggested by Withington (1980) and mentioned previously: duplication, lack of standards, and poor control. On the other hand, if DP encourages UDA by providing tools as well as training and assistance to users, UDA will emerge as one form of DDP, wherein application development responsibility, for certain types of applications, is distributed. And, as in the case of the more conventional view of DDP, Kaufman's statement would apply: it creates an environment that is responsive "to local and immediate needs, viewed emotionally, as OURS rather than THEIRS" (Kaufman, op.cit.). Also applicable would be the previously mentioned addendum: keeping overall control in corporate DP's hands.

From the above discussion, it appears that the main difference between the conventional view of the centralization - decentralization - distribution argument and the concept of UDA is one of unit of

analysis. The unit of analysis of the centralization vs. decentralization controversy is a company division, while the unit of analysis of UDA is the individual end user (or maybe the user department). While UDA involves issues that are not part of the centralization vs. decentralization discussion, such as the issue of having non-DP professionals develop applications, many of the managerial issues are common to both.

2.3.2 Data Administration

Literature on UDA stresses the importance of the availability of data if users are to be able to develop the applications they need. Data security and integrity have been identified as managerial concerns related to users developing their own applications. It is the role of Data Administration (DA) to ensure data availability while exercising proper control. This section gives an overview of the concept of DA and indicates its relevance to UDA.

Managing Data as a Corporate Resource

The idea that "data are a valuable resource, to be managed like any other resource" has received much attention during the past decade (Nolan, 1973; Secrest, 1975; Diebold, 1979; SHARE, 1980; Huff, 1981). This idea originates from problems created by the traditional DP environment.

In the traditional DP environment, data are collected, coded, and stored in files, with regard to the specific requirements of a program

or group of programs. It often happens that the same data item is contained in different files, because it is used in different applications; this is the problem of data redundancy. According to Nolan (1973) the most serious consequence of data redundancy becomes apparent when a data item has to be updated. All the files that include it should be updated. As Nolan puts it: "In an EDP department of any size, it is virtually impossible to update all the redundant files and reports in systematic and synchronized fashion" (p.104). The result is inconsistency in the reports produced by the different applications.

The second problem is the lack of flexibility of such an environment (Martin, 1982). If management wants information that requires data items which are located in different files, with different data structures, it might be very difficult to fulfill the need. As put by Secrest (1975) and emphasized in IBM's Business Systems Planning (BSP) methodology (IBM, 1981b), the traditional DP environment has a "vertical picture of data interaction within their application systems" (Secrest, 1975, p.648). This vertical picture was, to a certain extent, satisfactory for the development of independent, operational control types of systems. However, in several organizations, the core of these operational control systems have already been developed, and higher levels of management begin to ask for management control or strategic planning types of information. This requires the crossing of functional boundaries, that is, "a complete and accurate horizontal view of data interaction across the application systems" (Secrest, 1975, p.648).

The third problem occurs when changes have to be made to the structure of a file used by more than one program. In such cases, all the programs using the same file have to be changed. This problem directly affects DP, because it requires time and effort. It also affects users. A change that is considered minor, from a user point of view, may take months for DP to complete. This delay appears to the user as being way out of proportion to the magnitude of the change he requested.

The ideal solution to these problems has two dimensions. First, from a managerial point of view, data should be considered as a corporate resource, and a special function, namely, Data Administration, created to manage it. Second, from a technical point of view, the ideal solution would be (1) that there exists a single instance of each data item (i.e. eliminate data redundancy), (2) that cross-reference of data be made easy (i.e. achieve flexibility of the data environment), and (3) that changes in data structure and organization do not impact application programs (i.e. achieve data independence). Data independence implies that one should be able to make changes to application programs without having to be concerned with data structure. This ideal technical solution can be achieved via a corporate data base, and the related data base management system (DBMS).

Data Administration (DA) should be distinguished from Data Base Administration (DBA). DA is a corporate function whose responsibility is to provide resource management for the most recently recognized

corporate resource, data," while DBA's responsibility "for the design, performance and integrity of the physical data-bases is considered to be a technical responsibility delegated from DA" (SHARE, 1980, p.1). According to this definition DA pertains to the managerial aspects of data resource management, while DBA pertains to its technical aspects.

SHARE provides a description of the objectives of DA: (1) to maximize data availability, (2) while exercising control. Secret (1975), SHARE (1980), and Huff (1981) suggest activities which have to be undertaken by DA; these activities include the following.

Data planning must take place in order to determine what data are required, for which corporate activities they are required, what relationships exist between data items, and so on. Ultimately, data planning requires that a conceptual data model of the organization be built, that is, a representation of all the organization's computerized data items, and of relationships between these data items (Huff, 1981). Methodologies have been developed to perform conceptual data modeling; the entity-relationship (ER) approach (Chen, 1976) is one of these methodologies, and is the most frequently used in practice for conceptual data modeling (Talbot, 1982).

An inventory of data must be kept, in order to identify which data items are already available, where they are, how they are maintained, who uses them and who cannot use them, and so on. DA is also responsible for enforcing data privacy by controlling the acquisition and maintenance of data, and ensuring data security and integrity. If

conflicts arise among users, regarding the right to use the data, definitions of data items, etc., it is the responsibility of DA to mediate, "and in some cases perhaps to arbitrate" (SHARE, 1980, p.6). DA is also responsible for the standardization of items such as codes, formats, documentation, and so on. According to SHARE, standardization is extremely important to help assure the availability of data.

The practice of DA is either nonexistent or is in early stages in many organizations (McCririck and Goldstein, 1980; Huff, 1981). DBA, the technical dimension of managing data as a resource, is more widespread. The Data Base Administrator is responsible for designing and maintaining the data base structures (Martin, 1977) [9]. Schussel (1975) suggests three main areas of responsibility for the Data Base Administrator.

First of all, as custodians of the DB [Data Base], they [the individuals in charge for DBA] are responsible for the definition, control, and uniformity of the basic data itself. They are normally involved in the development of the data dictionary and the building of the DB on a computer file.... The second primary area of concern for the data base administrator is data structures, file organizations, and access methods that are used by the DBMS in storing and retrieving information from the DB.... The third broad area of responsibility for the data base administrator is the DBMS itself.... The DBMS must be "generated," must interface to the operating system, and must have a set of standards for proper interface with application programs. The maintenance of the DBMS also falls to the data base administrator. (Schussel, 1975, pp:671-672)

While, in the case of DBA, the importance of a good technical background is stressed (Schussel, 1975), for DA emphasis is put on the knowledge of how the organization "conducts business" (Secret, 1975;

Huff, 1981).

The main tools of DBA are the DBMS and the data dictionary. The primary function of the DBMS is to "serve the application programs by executing their data operations" (Martin, 1977, p.82). That is, the DBMS stands between the application programs and the data base. The data dictionary is, according to Date (1977), "the single most important DBA tool." The dictionary contains data about data (or, as termed by SHARE, metadata). This includes data definitions, mapping definitions, authorization checks, validation procedures, cross-reference information, and so on. In order to be useful, the data dictionary must be kept up to date (Date, 1977) and must be integrated with "the function that creates the data base" (Schussel, 1975).

Query languages are used by users of the data base (application programmers, systems analysts, or end users) in order to access and/or manipulate the data located in the data base. The queries handled by such languages are sometimes very complex and may involve several logical operations. Several commercial query languages exist. A few of the most well known (or well publicized) are: GIS (IBM), EASYTRIEVE (Pansophic), STAIRS (IBM), QUERY-BY-EXAMPLE (IBM), ON-LINE ENGLISH (Cullinane). Some of these languages are strictly query languages (QUERY-BY-EXAMPLE, STAIRS, ON-LINE ENGLISH) while others also have report generating capabilities (GIS, EASYTRIEVE). Martin (1982, pp.21-24) provides a list of software tools, wherein he indicates the data query possibilities, as well as other features of each tool.

Data Administration, Data Base Administration, and UDA

The importance of the DA concept, and of the related DBA concept and technology is stressed in the literature on UDA. The situation at APEX (Stork and Bland, 1980) illustrates some of the problems that may arise when UDA is implemented in the absence of a DA function. As Stork and Bland report, APEX does not have a central function that would help users in locating the source of the data they want to access. It happens that the members of a department's informal support groups assist other users in that task. However, such informal groups do not exist in all departments. When he chooses to handle the extraction process himself, the users must:

- 1) Personally identify an individual within the organization who has access to, or knowledge of the data, or
- 2) Personally identify all data files that contain the necessary information.
- 3) Set up necessary procedures to get the data transferred to the TIMS system, [and]
- 4) Personally insure that the data transfer occurs at the scheduled time. (Stork and Bland, 1980, p.106)

The above implies that much effort is required from users to get the data they need. At APEX, the lack of a central DA function also results in much duplication of data. Several users may extract, almost exactly the same files or portions of files. Yet, users tend not to "exchange" files or information about the data they are using. Stork and Bland report one user's response when asked why he did not use data already extracted by other users: "It would be too much trouble to try

to coordinate all that. It really is not my job" (p.107). Rockart and Flannery advocate that a formal central group should be responsible or assisting users in locating their data, obtaining them, and determining the quality of the data. That is, they advocate DA as a support group for UDA.

One of the responsibilities of Information Centre staff members, in IBM's view, actually consists of that which is advocated by Rockart and Flannery. Yet, according to Rosenberger (1981b), such a group must have appropriate tools. He reports the experience of an IBM customer, who "spent almost half of the total user support time locating and accessing the data for the user. The comprehensive implementation of the Data Dictionary greatly alleviates this" (Rosenberger, 1981b, p.920).

According to Martin (1982), and as discussed in Section 2.2.1, data bases and DBA have a "vital role" to play in the context of UDA, namely, to avoid duplication of data, fragmentation of applications, to facilitate maintenance, and so on. Data base query languages also have an important role to play in facilitating data access to users. However, for UDA, the most relevant dimension of all these tools (DBMS, query languages, data dictionaries, etc.) is their degree of ease of use, or user friendliness. This is the essence of the topic addressed next.

2.3.3 Human-Computer Interactions

The literature on human-computer interactions is relevant to this study mainly with regard to the characteristics required from software tools that serve to make these tools user friendly. This is not to say that hardware considerations do not have importance in the context of UDA. However, most hardware considerations, such as keyboard selection, CRT versus hard copy, standard keyboard versus light pen, audio input and output, etc., are outside the scope of the present study.

To some authors, a dialogue between the user and the computer should take place in the user's natural language (Codd, 1978; Waltz, 1978). In a discussion on the appropriateness of natural language as a query language, Zloof (1978) summarizes Codd's and others' arguments as follows: "the only way to entice such a user [i.e. casual] user to interact with a computerized data base is to permit him free use of his native tongue" (Zloof, 1978, p.32) [10]. Research to develop such languages has been ongoing for several years; ELIZA (Weizenbaum, 1976), RENDEZVOUS (Codd, 1978), and PLANES (Waltz, 1978) are instances of such research efforts. However, the adequacy of natural language as a means of communication between user and computer is often disputed. According to Martin (1973), while an English-like dialogue is in theory the most natural man-machine interface, the ambiguity of the English language makes it inappropriate when a user's commands must be precisely interpreted. Smith claims that "to design computer programming languages which look like English...is not possible without imposing

undue constraints on languages terms that have to reflect underlying control structures" (Smith, 1980, p.29).

Schneiderman (1980) identifies five problems that natural language systems must overcome:

1. Unrealistic expectations of the computer's power...
2. Attempts to request information not contained in a database, thus wasting time and effort, while increasing frustration...
3. By allowing users to use natural language without training we allow the ambiguities of English syntax to pollute the query process...
4. Typical users may not be aware of the semantics of question asking...
5. The overhead of creating and maintaining a natural language interface will always be larger than concise query languages or a menu selection process.

(Schneiderman, 1980, p.208)

There exist characteristics that a dialogue should have so that a non-DP trained user feels comfortable using it. Bennett (1972) summarizes primary ease-of-use criteria proposed by R. Miller [11].

These criteria are:

- (1) length of time for user population to learn to reach a level of competence;
- (2) proportion of relevant population which can learn to use the tool in a given time;
- (3) time required to reach "communication" level...;
- and (4) user attitude toward employment of the tool. (Bennett, 1972, p.169)

Schneiderman reviewed the literature on interactive systems design and presented design guidelines proposed by eight different authors (1980, pp.250-256). Commonalities exist among the eight sets of guidelines. Emphasis is put on the importance of providing users with clear, meaningful error messages. Systems should also be protective, in the sense that they should prevent serious accidents such as overwriting a file. System flexibility is a characteristic part of the majority of the eight sets of guidelines. Flexibility of a system includes adaptability of the system to different degree of sophistication of users, the possibility of using short forms for keywords, and so on. Simplicity of the system is also emphasized. In the context of these guidelines simplicity pertains to minimization of memorization, minimization of the number of keywords, and the presence of escape features, that is, the possibility for the user to stop or redirect the interaction when he wants to do so. Powerful features, that is, commands that do much in a single step, are also recommended.

Zloof (1978) provides a set of important human factor considerations for designing a data base query language which is to be user friendly. He suggests that "the extent to which a language satisfies these requirements will determine the degree of user-friendliness of that language"(p.38). The nine considerations suggested by Zloof are outlined below, since they appear to encompass most of the recommendations made in the eight sets of guidelines previously mentioned. The considerations are:

1. Minimum concepts are required to get started.
2. Minimum syntax.
3. Consistency.
4. Flexibility.
5. Not sensitive [to small changes in the query].
6. Easy to extend and modify.
7. Minimum exception rule.
8. Easy detection of errors.
9. Unified languages.

(Zloof, 1978, pp.38,39)

Apart from providing key considerations in the design of dialogues, research in the area of human-machine interactions is also relevant to the present study with regard to support that should be provided to users. Eason, Damodaran, and Stewart (1975) conducted a survey of naive computer users [12], in order to determine why these individuals use computers, and the main issues associated with their interaction. The results of the study suggest that even if a user is well trained with a system (or with a tool, in the present context), learning most often takes place in actually using the system. In order that this learning takes place, users must be supported by good documentation and human advisors. T. Martin, Carlisle, and Tr en (1973) also emphasize the importance of "multimedia training and support via manuals and human trouble-shooters." Another critical issue Eason et al. identified is related to the concept of user friendliness, or ease of use, of the tools. The study findings suggest that, while some degree of standardization exists with regard to computer hardware, this is not the case for software. "As a result, as the user moves from system to system, he has to learn a new 'language'" (Eason et al., 1975, p.104). This finding corroborates Zloof's recommendation for

unified languages.

To sum up, two main lessons were learned from the human-computer interactions literature. They are the following. First, the software tools made available to non-DP users must be user friendly. While this requirement had already been discussed in Section 2.2, the human-computer interactions literature provides a set of considerations for determining the degree of user friendliness of a tool. Second, the literature stresses the importance of support provided to users via human advisors.

2.3.4 Organization Change

The relevance, to MIS, of the literature on organization change is widely recognized, mainly in the context of the implementation of information systems, as well as the implementation of Operations Research/Management Science (OR/MS) models (Sorensen and Zand, 1975; Ginzberg, 1975b; Keen and Scott Morton, 1978; Narasimhan and Schroeder, 1979). The rationale for studying implementation as a change phenomenon is the following:

OR/MS practitioners are constantly attempting to bring about changes in client organizations through their models and personal involvement in decision situations. The personal involvement between the scientist and the manager may be characterized as an "influence relationship." The nature of this relationship is such that it may bring about changes in the perspectives of the decision makers and in their decision-response patterns. (Narasimhan and Schroeder, 1979, p.63)

This approach focuses on the relationship between the consultant (DP professional or OR/MS specialist) and the client (end user). The consultant typically initiates the change, and the client is affected by it.

The case of UDA is somewhat different, with regard to two main aspects. First, while the implementation of an information system affects the user only, the introduction of UDA affects both DP and the user community. Whereas the conventional role of the user is to stay relatively remote from the development process, UDA, by definition, makes the user the application developer. Consequently, some aspects of the task of the user are changed. On the other hand, in the traditional DP environment, the DP department controls the delivery of information to users, advocates the need for systems development standards and methodologies, data editing, audit trails, and so on. UDA implies that DP no longer has complete control over the delivery of information, that applications will be developed by people who generally have no DP background and who are not likely to use standard methodologies. To sum up, it appears that the major change UDA brings to DP in general, and to each DP professional in particular is not a change in task but a change in image. The second difference resides in the source of change. The role of the DP professional as the initiator of change as well as of the change agent is generally agreed upon. Previously, it was shown (Section 2.2.2) that, in the case of UDA, the initiator can either be DP or the user community, and that either one of them can resist the change necessitated by the introduction of UDA.

The relevance of the organizational change literature to UDA is that it sheds light on the understanding of the dynamics of the introduction of UDA in an organization. The following paragraphs focus on three key issues: resistance to change, DP management readiness for change, and the implementation of change.

Resistance to Change

Every change in one part of a system creates a reaction in another section. Even when you know how to produce a desired effect and the benefits far outweigh the risks, there is one more obstacle to overcome before principles of change become policy - POWER....To be effective as a change agent, one must recognize two sources of power: one's own and that of all adversaries. Then the task becomes the difficult one of changing the balance of power in your favor. (Zimbardo, Ebbesen, Maslach, 1977, p.193)

The above supports Martin's (1982) claim that the introduction of UDA may have to face much resistance in many DP departments. One explanation Martin provides for such a resistance is that senior DP managers can be alarmed by the potential consequences of the change, and "sometimes envision their empire dwindling if end-user departments learn to create their own applications" (Martin, 1982, p.76). However, change as being a threat to individual power bases, is not the only explanation for this resistance to change. Moreover, DP is not the only "actor" who might resist change. It might happen that users also resist the introduction of UDA.

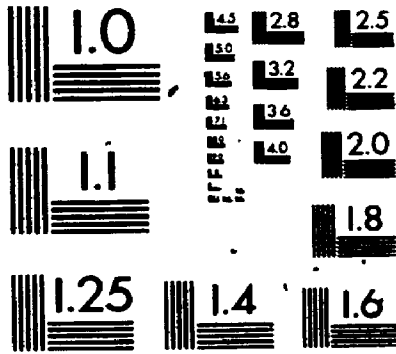
Lorsch and Kotter explain resistance to change by the following:

People generally seem to resist change that threatens their psychological contract....If the change, in effect, takes something of importance away without a compensating change in the psychological contract, people will probably resist vociferously, or by just dragging their feet. Persons may also resist changes because the new behavior violates the psychological contract by requiring them to act in ways beyond their personal capabilities. It may require a set of skills which they find it impossible to develop. Also, persons may find that the new behavior is inconsistent with their personalities and established ways of interacting with others. Another reason why resistance to change is so frequent is that a request for change in behavior often creates increased uncertainty for the individual. (Lorsch and Kotter, 1976, pp.3-4)

The reasons for resisting change, as proposed by Lorsch and Kotter, apply directly to the introduction of UDA in organizations. The lack of compensating gain and the increase in uncertainty effectively explain resistance from the DP community, while the perception that the change demands something people cannot do pertains to resistance from the users.

If UDA is perceived by DP as taking away some power, without bringing any benefits, resistance is likely to occur. UDA also creates uncertainty for DP professionals. As put by Martin, the theme of UDA "must cause programmers to wonder about their future jobs." That is, in addition to changing the image of the DP professional, UDA can also create uncertainty with regard to the extent to which they are indispensable. If UDA was the only threat programmers had to face, the degree of uncertainty created by it might in many situations, be too low to be worth mentioning. However, UDA is not the only threat faced by programmers. Kraft (1977) argues that the introduction of modern programming techniques (such as those discussed in Section 2.1.1) is

2



perceived by many DP professionals as a threat to their independence. In such situations, UDA may have the effect of increasing the uncertainty of the DP professional environment.

From a user point of view, UDA may be perceived as a change "requiring a set of skills which they find it impossible to develop," and consequently generate resistance. Martin claims that "a high proportion of the people who ought to be using computer terminals are afraid of them." This fear extends to both the hardware and the software. Lorsch and Kotter also mention that "persons may find that the new behavior is inconsistent with their personalities and established ways of interacting with others." To some people, interacting with a computer through a terminal projects an image of technically oriented activities. Some users might not want to project such an image of themselves, and are therefore likely to resist the introduction of UDA.

The previous paragraphs imply that resistance to change always occurs when organizational change takes place. While some argue that "successful innovation occurs only after initial resistance has been worked through" (Klein, 1976, p.120), others claim that in some instances, people "not only accept change, but they are a positive force in helping to bring it about" (Lorsch and Kotter, 1976, p.2). In the case of UDA, the latter view is much more representative of reality. The experiences of firms that have implemented UDA strongly suggest that, in general, user communities do not resist the change that takes place, and often are the prime movers for the change.

DP Management Readiness for Change

Literature on organizational change helps explain, from a conceptual point of view, why and when a DP department would introduce UDA. Zaltman, Duncan, and Holbeck (1973) claim that a performance gap [13] must exist and be recognized by the organization before action be taken toward change. "Given the assumption that the organization is aware of the gap, a need can be said to exist" (p.3). However, for the perception of the gap to create a need for change, the following assumption is necessary: "The performance gap is perceived as having significant adverse consequences for the organization if the gap is not narrowed or bridged" (p.3). Transposed to the context of DP, Zaltman et al.'s assumption suggests that a DP department has to (1) perceive a discrepancy between its actual performance and what its external environment (users) demands and (2) this discrepancy must be perceived as having significant adverse consequences. Consequently, it might be possible that the user community is unhappy with DP, that DP is aware of it, but it is suggested that DP will not be ready for change until the consequences of the discrepancies are perceived as being threatening.

While Zaltman et al. focus on the role of pressures from the external environment in bringing about a need for change, Greiner (1967) also emphasizes the importance of internal pressures for creating such a need.

In the case of DP, internal pressures pertain mainly to the issue of DP professionals' productivity. Instances of such pressures are

increasing programming costs, increasing DP application backlog, increasing maintenance load, and so on.

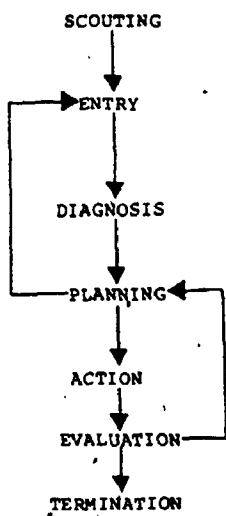
Greiner surveyed several change studies in order to understand the dynamics of successful change. He claims that: "Until the ground under the top managers begins to shift, it seems unlikely that they will be sufficiently aroused to see the need for change" (p.126). The results of the survey he conducted indicate that "the success patterns suggest that strong pressures in areas of top management responsibility are likely to provoke the greatest concerns for organization change" (p.126).

Looking at the DP department as an organization, the above suggest that for DP to be ready for change, the senior DP manager has to perceive strong pressures, both from DP itself and from users. Moreover for the change to be initiated, the gap between DP's current performance and what its environment (both internal and external) demands must be perceived, by the senior DP manager, as having significant consequences.

The Implementation of Change

A large number of models have been proposed to explain the process of change and/or to assist in implementing change, so as to make it successful. Three models of planned change are outlined in Table 2.3. Notice that from one model to the other, the number of stages of the change process, as well as the importance given to each stage, vary. For instance, while Beckhard and Harris identify the first stage as

TABLE 2.3
MODELS OF CHANGE



(Kolb-Frohman, 1970)

1. Creating an awareness of the need for change.
2. Making a diagnosis of the situation which creates the need for change.
3. Communicating the change to those affected.
4. Monitoring the change and making adjustments as they seem appropriate.

(Lorsch and Kotter, 1976)

1. Diagnosing the present condition including the need for change.
2. Setting the goal and defining the new state or condition after the change.
3. Defining the transition state between the present and the future.
4. Developing strategies and action plans for managing this transition.
5. Evaluating the change effort.
6. Stabilizing the new condition and establishing a balance between stability and flexibility.

(Beckhard and Harris, 1977)

being the diagnosis of the present condition. Lorsch and Kotter identify the creation of an awareness of the need for change as preceding the diagnosis stage. The Kolb-Frohman model still puts more emphasis on the pre-diagnosis stages. It suggests that scouting (i.e. mutual "sizing up" between client and change agent) and entry (i.e. the negotiation of a "contract" between the two parties and the development of a relationship based on trust) have to take place before a diagnosis can be made.

Notwithstanding the value of a discussion on the differences between the three models of Table 2.3, it appears that their main similarity is more relevant to the present context. All three models are based on the Lewin-Schein (Schein, 1969) conceptual scheme, which involves three phases, namely, unfreezing, changing, and refreezing.

Those phases are defined as follows.

Unfreezing. Unfreezing is overcoming resistance to change by encouraging dissatisfaction with the current behavior in order to unlearn this current behavior and create a desire to learn new behavior. It involves reducing the defensiveness to learn so the individual will search and explore the environment for new and improved ways of behavior.

Changing. Changing is exposure to new information, attitudes, and theories in order to achieve new perceptions and learn new behavior patterns. It involves gathering new information, considering and interpreting new and existing information, and developing alternative courses of action, which could lead to learning new behavior patterns. In this process a method for choosing among alternatives may need to be developed or perhaps modified.

Refreezing. Refreezing is reinforcement, and confirmation or support of new behavior. It requires integrating new responses into the total personality and attitude systems for the individual and integrating new responses into ongoing significant relationships for the organization. Refreezing can be achieved through positive feedback and is necessary for the resulting stability of the new situation. (Sorensen and Zand, 1975, pp.218-220)

Transposed to a situation where a DP department would consider the introduction of UDA, the model suggests the following. DP (or its delegated change agent) must prepare the user community for accepting the new behavior (unfreezing). The experience of firms with UDA (reported in Section 2.2.2) suggests that in many cases, this stage of "encouraging dissatisfaction with the current behavior," might not be necessary. It appears that an important proportion of the user community is already dissatisfied with the "current behavior" [14].

The second phase, changing, takes place when DP makes programming tools, training, support, etc., available to users. Finally,

refreezing occurs when the new behavior is reinforced. In the case of UDA, behavior reinforcement would have several dimensions, a few of which are suggested here: DP should (1) maintain a good quality of support after users have been trained with UDA, (2) attempt to maintain quality of computer system performance (response time), (3) motivate users via annual conferences, bonuses, etc. (Martin, 1982), (4) concentrate on early adaptors among users and demonstrate that the early adaptors are doing something valuable (Martin, 1982).

Both the Lewin-Schein and Kolb-Frohman models have been used in research on MIS implementation. Sorensen and Zand (1975) used the Lewin-Schein scheme in their study, while Ginzberg (1975b) used the Kolb-Frohman model. In both studies, it was found that refreezing (or termination) was the phase which was the most closely related to success of implementation. In the case of Ginzberg's study, "in looking at the stages one at a time, the only clear difference was at termination...There were no clear and consistent relationship between success and the scores on other stages" (Ginzberg, 1975b, p.249). In the case of Sorensen and Zand's study the three stages of Lewin-Schein's model were found to be significantly related to success of implementation. Yet, the indices used to measure the degree of refreezing had the highest correlation coefficients with the variable implementation success.

In summary, three dimensions of UDA were discussed in the light of the organization change literature. First, reasons for which either DP professionals, or users would resist change were suggested. It was also

indicated that resistance to change is not a necessary component of change; that is, changes may occur where people not only accept the change which takes place but are also prime movers. Second, the importance of DP readiness for change was discussed. This discussion mainly pertained to some necessary conditions, in the absence of which DP is not likely to introduce tools and support for UDA. Those conditions indicate that the senior DP manager has to perceive a gap between the demands of the environment of DP (both internal and external) and DP's performance. Moreover, he must perceived the gap has having serious, adverse consequences. Finally, the usefulness of some models of planned change for helping in the introduction of UDA was discussed.

2.4 SUMMARY

This chapter first discussed the phenomena from which UDA originates. Those phenomena were classified in two categories, namely, DP problems in the context of systems development and users' problems in the same context. Second, the UDA literature was reviewed, and presented as follows: (1) essays on UDA, (2) reported experience of firms with UDA, and (3) research-based literature. Finally, literature in other relevant areas was surveyed. Those areas are (1) centralization vs. decentralization of DP, (2) data administration, (3) human-computer interactions, and (4) organization change.

The initial research model, which organizes the previously reviewed literature, will be discussed in the next chapter.

FOOTNOTES

[1] McLean refers to perfective maintenance, as defined by Lientz et al. (1978). The different types of maintenance have previously been defined in section 2.1.1.

[2] According to Martin, "if the users can learn to be comfortable with the product and carry out useful work with it in 2 days and not necessarily need to return to class on it later" then the product has passed the 2-day test.

[3] According to Comper's (1980) presentation, the concept of having users develop their applications themselves was introduced after data retrieval services introduction: Data retrieval services were responsible for developing the applications for users. Rosenberger (1981b) gives a definition of data retrieval services.

[4] See 2.1.1 for a presentation of Alloway's findings.

[5] Lauren Stork (Stork and Bland, 1980) is Lauren Flannery (Rockart and Flannery, 1981).

[6] The phrase "end user computing" used by Stork and Bland, and Rockart and Flannery encompasses a broader population of users than the acronym UDA does. As it will be seen when APEX user profile is presented, the target population of users included users who do not develop applications. UDA, by definition, does not include such users.

[7] While a more recent, six-stage theory (Nolan, 1979), "supersedes the four-stage concept" presented by Gibson and Nolan (1974), Rockart and Flannery base their discussion on the four-stage concept. They argue:

Recently, Nolan has added two additional stages to the model. He has been appropriately challenged as to the usefulness of this extension. It has been pointed out that his original stage theory is, in effect, a particularization of the general phenomenon of the "learning curve..." The purpose of this paper is to present the results of some ongoing research which strongly suggests that we have yet another managerially significant multi-stage "learning curve" taking place today in the information systems field.

Consequently, in order to remain consistent with Rockart and Flannery's approach, the original version of Nolan's stage theory is presented in the body of this section. However, the main features of the six-stage concept are summarized herein.

Nolan (1979) identifies six stages of DP growth: Initiation, Control, Integration, Data administration, and Maturity. In the revised version of the growth theory, the first three stages remain relatively unchanged from the description given in the body of this section. Initiation starts with the introduction of the computer and

is characterized by the development of applications in one main area. Contagion is characterized by an increased demand for DP services and little or no control. Stage 3 occurs when top management becomes concerned with the rapid, uncontrolled growth of DP. The main characteristic of the Control stage is an effort to control every aspect of the DP activity in the organization. The Control stage appears to be the era of standards and methods, and is also characterized by the introduction of user chargeout. Because of these efforts put on control, DP appears to users as not doing much for them, while making them pay for DP services. According to Nolan, during this stage many users are frustrated and "give up on data processing." Stage 3 is also characterized by a transition in management orientation. Nolan claims that up to "sometime in stage 3," the main concern is the management of the computer while, "at some point in stage 3, there is a transition to management of data resources" (p.116). Nolan explains this shift by the problems created by the fact that "base-level [operation control] systems cannot support higher-level systems [management control and strategic planning]" (p.118). The end of Stage 3 is characterized by the introduction of the data base and of the data communication technology, as well as by a consolidation, from a technological point of view, of the services provided to users.

Stage 4, Integration, is characterized by an effort, from DP, to make users benefit directly from the data base technology. This effort is translated by the spreading of interactive terminals and the availability of assistance for using the data base technology. Users perceive the value of it, and "they virtually demand increased support and are willing to pay pretty much whatever it costs" (p.120). According to Nolan, Stage 4 witnesses a growth rate in DP's budget as dramatic as it was in Stage 2.

Stage 5, Data administration, is characterized by the implementation of the concept that data is a corporate resource, while Stage 6 is the Maturity stage.

[8] Shepherd (1978), Van Rensselaer (1979) and Withington (1980) provide concrete examples of DDP.

[9] It is relatively difficult to clearly distinguish the responsibilities of DA from those of DBA. The reason is that in the literature, the two terms are often used interchangeably. For instance, in the index of Martin (1977), the two terms refer to the same pages of the book, where DBA only is used.

[10] Codd defines a casual user as: "...one whose interactions with the system are irregular in time and not motivated by his job or social role. Such a user cannot be expected to be knowledgeable about computers, programming, logic, or relations." (Codd, 1974, p.179)

[11] Bennett refers to:

Miller, Robert B., "Archetypes in Man-Computer Problem Solving," IEEE Transactions on Man-Machine Systems, Vol.10, No.4, December 1979,

p.219-241.

[12] Naive users are defined by Eason et al. (1975) as "individuals working within an organization, who are not experts in computer technology but who use the computer to assist them in their work." (p.92).

[13] Zaltman et al. (1973) define performance gaps as follows: "Performance gaps are discrepancies between what the organization could do by virtue of goal-related opportunity in its environment and what it actually does in terms of exploiting that opportunity." (p.2)

[14] Here the relevance of the diagnosis phase suggested by the three models of Table 2.3 becomes clear. That is, the result of the diagnosis phase might very well be that unfreezing has already taken place.

CHAPTER III

INITIAL RESEARCH MODEL

This chapter presents a model of the factors of success for UDA. The model serves two purposes. First, it helps organize the literature reviewed in Chapter II. Up to now, in the UDA literature, no effort has been made to develop a conceptual framework that would (1) organize the literature in conjunction with other relevant areas, (2) develop conjectures with respect to the relationships between the different elements of the literature, (3) define success of UDA, and (4) suggest conjectures with regard to the relationships between the different elements of the literature and success of UDA. The model presented here is then an effort to integrate the existing literature around the foregoing points. The second purpose of the model is to provide direction for the present study. It will be used as a basis for investigation, and will be refined based on the results of the field work.

Table 3.1 lists the independent variables of the model, and indicates the reasons for their inclusion in the model. The independent variables are user friendliness of the tools, quality of support provided to users, control, DP readiness for change, strategy, maturity of DP, and user attitude toward UDA. Each of these variables will be discussed below. However, in order to make their importance clear, the independent variable, that is, success of UDA, is defined first.

TABLE 3.1

INDEPENDENT VARIABLES

User friendliness of the tools

Explicitly suggested by UDA literature and supported by man-computer interaction literature

Support to users

Control

Explicitly suggested by UDA literature

DP readiness for change

Inherent to change nature of UDA and supported by literature on organization change

Strategy

Deduced from UDA literature

Maturity of DP

User attitude toward UDA

Suggested by literature on organization change and MIS implementation literature

3.1 THE DEPENDENT VARIABLE

This study defines success of UDA from two perspectives: from the end user's point of view and from DP's point of view. In doing so, it is not implied that these perspectives oppose each other, only that users and DP have different ways of determining whether UDA is successful, as related to their own concerns and interests. Success from DP's point of view is measured here by the degree of diffusion of

UDA, and by productivity improvements, the main ones being decrease in the backlog and decrease in the maintenance load. Success from the user's vantage point is measured by user satisfaction with the concept of UDA and with the organizational context in which it takes place.

3.1.1 Success for DP

The model presented here suggests that, from DP point of view, UDA will be seen as successful if it has a positive impact on DP productivity. Chapter II presents Rosenberger's (1981b) list of 10 "productivity factors" due to Information Centres (IC). However, two of these productivity gains are most frequently cited (either explicitly or implicitly) in the UDA literature. These two productivity gains are: the reduction of the DP application backlog and the decrease of the DP maintenance load. With respect to maintenance, Rosenberger claims that reduction of the maintenance load is "the most direct and obvious effect of the Information Center within DP" (p.921). McLean (1979), and Holmes' (1980) also identify maintenance reduction as the most probable and most important consequence of UDA, and Martin (1982) makes the same suggestion.

Reduction of the DP backlog is the second most often mentioned benefit of UDA, from a DP point of view. As mentioned in the previous chapter, Rosenberger states that some firms have observed a decrease of their application backlog after introducing an IC. Martin states that adopting UDA requires a "strong desire to eliminate the long backlog" (Martin, 1982, p.241).

Yet, one must be cautious in measuring both the decrease in the backlog and the decrease in the maintenance load. With regard to the backlog, there are two reasons for caution. First, there exist many different types of computer-based applications, and UDA may not decrease the requests for all these types. At this point, a digression is necessary. Following McLean (1979), three different types of applications are defined here: personal applications, departmental applications, and corporate applications. Personal applications are designed to serve the needs of an individual. "They draw upon capabilities, facilities and data that are already in place" (p.42). Departmental applications "provide the reports, both routine and special, the queries, the analyses, and the many other items of computer-based data that form the backbone of a department's management information system" (p.43). Corporate applications "are designed to meet external, as well as internal requirements" (p.43). They may involve data from several departments and are generally very large systems. The literature suggests that applications developed by users will generally be of the personal or departmental type. For instance, McNurlin states that: "We have not heard any proponent of end user programming suggest that end users take over the programming work on the larger, more complex applications currently being done in data processing" (McNurlin, 1981, p.5). Also, according to IBM Canada's "rules" for implementing an IC, users are not allowed to develop corporate applications (applications that involve several departments; see Grindlay, 1980). Consequently, what has to be looked for in measuring the reduction in the backlog is not the entire backlog: the overall number of applications waiting to be developed may be the same

or even higher than before the introduction of UDA. What has to be investigated is a change in the composition of the backlog. It would be expected, as suggested by Rosenberger (1981b) that UDA would "skim off the top" of the backlog the ill defined, one-shot type of applications, while leaving only the larger "production" systems.

The second reason for caution in the measurement of change in the backlog is that other events, apart from UDA, may have taken place in an organization during the period over which change is measured. For instance, DP may have hired more programmers or analysts, may have implemented some of the modern programming techniques discussed in Chapter II, or both. The backlog may also have decreased because users were so dissatisfied with the service provided by DP that they have given up requesting applications or "have left the queue." Consequently, when measuring change in backlog, it will be necessary to take such factors properly into account.

Caution is also required in measuring the decrease in the maintenance load. First, the type of maintenance has to be identified. The literature suggests that UDA is likely to have an impact on perfective maintenance (McLean, 1979) as defined by Lientz, Swanson, and Tompkins (1978) [1]. Yet, there still remains an issue. Even when the perfective maintenance load decreases, one can ask if there has been a real reduction of maintenance costs or if there has merely been a transfer of such costs to the user departments. Total costs for the organization as a whole might even be higher. Given the users relative lack of expertise with application development, it may take them much

more time to perform a given development task than it would take DP professional using the same software tool. Moreover, the code written by users may be less computer-efficient than code written by DP professionals. This issue has to be raised during the investigation.

The degree of diffusion of UDA is included in the model as a measure of success, from the point of view of DP. It could be argued that the mere fact that UDA is widespread in an organization might not be interpreted as a measure of success. DP managers would likely be more interested in the reduction of the backlog and of the maintenance load than in UDA diffusion per se. However, in order that backlogs and maintenance load decrease, at least some new applications have to be developed (or existing ones be enhanced), by users (all other things - number of analysts and programmers, programming techniques, etc. - being equal). The degree of diffusion of UDA can be measured, as suggested by Rockart and Flannery (1981), by the percentage of the firm's computer resources that are used by the user community in UDA activities. Alternately, one can determine the proportion of the potential population of user-application developers who have already started to develop applications.

3.1.2 Success from the User Point of View

In the initial research model, user satisfaction is used as a surrogate of success of UDA from the point of view of the user community. The use of user satisfaction as a measure of success or as a means of evaluating systems is relatively common in MIS research.

Nolan and Seward (1974) for instance employ user satisfaction as a surrogate for information utility. Powers and Dickson (1973), Ginzberg (1975a, 1975b), Lucas (1978), and McKeen (1982), used user satisfaction as a measure of success of a system, from a user point of view. As put by Ginzberg "essentially, if the manager is satisfied that a project met its goals, the project should be termed a success" (Ginzberg, 1975a, p.7).

This study identifies two main dimensions of user satisfaction: satisfaction with the concept of UDA and satisfaction with the organizational context in which UDA takes place. The first pertains to satisfaction with the fact that the user has the opportunity to develop applications himself, as opposed to having them developed by DP. Some potential consequences of UDA, for users, are increased productivity, timeliness in application development, reduction (or even disappearance) of the need to communicate application requirements to outsiders, applications more readily adaptable to changes, and so on.

Application development in general implies that new relationships develop between users (or user departments) and the organization's DP department. Such is also the case in the context of UDA. It is difficult for users to develop applications in complete isolation from DP. Often users will need to access data from files or data bases controlled by DP, will need access to a computer also controlled by DP, will need the assistance of DP professionals, and so on. Even if none of these needs exist (as would be the case when a user department has its own computer, its own files, and its own expertise - for instance,

the Factory Production Control System described by Martin, 1982) there still exist relationships with the DP department: DP may agree with the fact that users "do their own thing" or it may oppose this. This is why user satisfaction with the organizational context in which UDA takes place is used here as a measure of success. Users may be satisfied with the UDA concept, but still find that the context in which they have to do it is not satisfactory. They may also be satisfied with both, or dissatisfied with both. This second dimension of user-UDA success can be measured by directly asking users how satisfied they are with several elements of the organizational context in which UDA takes place.

3.2 THE INDEPENDENT VARIABLES

The proposed model identifies seven independent variables. Those variables were previously listed in Table 3.1. Some variables are explicitly suggested by the literature on UDA; these are: user friendliness of the tools, support provided to users, and control. The inclusion of DP readiness for change was suggested by the literature on organization change. Strategy and maturity of DP are variables that are not explicitly suggested by the literature on UDA, but their importance and role can be deduced from this literature. Finally, the importance of user attitude toward UDA was deduced from two sources, namely, the organization change literature and MIS implementation research. Each of these variables is discussed below.

3.2.1 User Friendliness of the Tools

"User friendliness of the tools" is an often used phrase in the UDA literature. The degree of user friendliness of a particular software tool depends on who the user is. As put by McLean (1979): "What is very logical and obvious to a DP professional can be very confusing and mysterious to the manager attempting to find out a critical item of information locked away somewhere in the system" (p.40). In this study, the degree of user friendliness of the tools is determined by asking users how they perceive the tools they use. The components of user friendliness that are to be employed are those proposed by Zloof (1978), as outlined in Chapter II [2].

3.2.2 Support Provided to Users

Studies show that user support is an important issue in the broad context of human-computer interactions (Eason, Damodoran, and Stewart, 1975; Martin, Carlisle, and Tren, 1973). Actual experience firms have had with UDA suggests that it is also important in this particular context, as does the APEX study (Stork and Bland, 1981). The present study proposes two important dimensions of support, namely, variety and availability of support. Variety of support pertains to the range of the support function. For instance, IBM (1981a) suggests that user support should include: formal training with the use of the tools, advice on the choice of the proper programming tool to use, assistance during the design of an application, debugging assistance, help with planning for future usage, and so on. Availability of support

encompasses issues such as: Are there enough resources (e.g. manpower) devoted to user support? When a problem occurs during development, is there much delay in obtaining assistance?

The present study attempts to determine the quality of support first by identifying, within DP departments, what type of support is provided to users, and second, by measuring users' perception of the quality of that support.

3.2.3 Control

Firms' experience with UDA, and with Information Centres in particular, indicates that some kind of control is imperative. One aspect of control is the determination of the type of applications users are allowed to develop. At IBM Canada, for instance, a program that would either generate data, update files, be shared by several departments, or be critical to the business has to be developed by DP (Grindlay, 1980). Another aspect of control that receives much attention in the UDA literature concerns data security and integrity (see Holmes, 1981; Comper, 1981). The use of resource charge-back techniques for maintaining control is also discussed (Holmes, 1981; McNurlin, 1981; Rockart and Flannery, 1981).

For the purposes of this study, it is the intent (with respect to UDA) of the control that matters, rather than the actual means of control. A DP department may establish control mechanisms in order to better manage its resources, ensure security, etc., while still being

favorable to the concept of UDA. In such a case, the intent of control would be non-UDA-restrictive. On the other hand, control mechanisms may be used so as to restrict UDA. Difficult access to the organization's computer(s), or users not allowed to buy computer time outside the organization would be instances of UDA-restrictive controls.

3.2.4 DP Readiness for Change

As discussed in Section 2.3.4, UDA brings change to both DP and the user community. The organization change literature suggests that the concept of readiness for change contributes to explain why DP would introduce UDA. Since it has already been discussed in Chapter II, the concept of DP readiness for change is only briefly reviewed below.

In the present study, DP is said to be ready for change when the senior DP manager perceives strong pressures, both from the user community (external environment) and from DP itself (internal environment). Such pressures may arise from many factors, some of which are listed below (as suggested by IBM, 1981a):

Pressures from the Internal Environment

1. Inability of DP to respond to small user requests, due to resources committed to larger projects.
2. Large proportion of DP resources devoted to maintenance.
3. Increase in programming costs.
4. Large backlog of applications.

Pressures from the External Environment

1. User dissatisfaction with DP services.
2. Increase in outside time-sharing usage.
3. Increase in "computer literacy" of users, combined with increase in user demands for accessing and manipulating their own data.
4. Increase in user demand for new services (e.g. problem solving, text processing and retrieval, etc.).

The above might not be unusual in modern organizations. However, the organization change literature suggests that, in order for DP to be ready for change, the gap between the demands of the environment and the current performance of DP has to be perceived as having significant adverse consequences for DP itself.

3.2.5 Strategy

While the variable strategy is not explicitly referred to in the UDA literature, its importance and its role are implicitly suggested. The use of the word strategy in the present context pertains to the mode of inception of UDA in an organization. Two types of strategies are identified here: a Pull strategy and a Push strategy. A Push strategy exists when DP initiates the concept of UDA. It may take place in several different ways: providing users with timesharing facilities; encouraging the acquisition, by user departments, of microcomputers for certain stand-alone applications; providing users with user-oriented tools; or establishing a support group of the IC type. A Pull strategy exists when users start developing their own

applications without being encouraged to do so by DP, and even, in the extreme cases, in spite of DP. The case of the Factory Production Control System (Martin, 1982) developed using MAPPER is an instance of a Pull strategy. There may exist mixed strategies, where there is a Pull from the user community, as well as a Push from DP. For instance, DP can provide in-house timesharing facilities to the user community, while at the same time, some users buy computer time outside. Such a situation may happen if there is a poor "fit" between what DP offers and what users demand.

As discussed in Section 2.2.2, one element of the "conventional wisdom" on UDA is that DP encourage and provide coordination to UDA. That is, that DP should adopt a Push strategy. Some argue that in the case of a Pull strategy (i.e. when DP does not provide coordination), DP will lose some control of corporate computing (Rosenberger, 1981b) and that negative effects such as "fragmented departmental data, poorly written user programs, and hardware that cannot be connected to a company's data network," may occur (McNurlin, 1981).

3.2.6 Maturity of DP

As in the case of the variable strategy, the UDA literature does not explicitly refer to the concept of maturity of DP. However, by assuming a certain degree of maturity, the literature implicitly suggests the relevance of the concept.

For instance, the claim that UDA makes corporate data readily available to users, assumes that an important portion of the operational control applications are already in place, so that there exists data to be made available to users. "Technological maturity" is also assumed. None of the examples of UDA given in Chapter II describe users employing key punch and punch cards [3], and the majority involved the use of data base technology.

The main reason for including maturity of DP in the initial research model is to explore the relationship between this variable and the variable strategy. As will be discussed in the following section, one proposition of the model suggests that a relationship exists between the variables maturity of DP and strategy. The concept of maturity of DP is discussed below.

The degree of maturity of DP in an organization is not necessarily proportional to the number of years the organization has been using computers to process its data. As stated by Ein-Dor and Segev (1978); "old established organizations which are run informally and intuitively may be very immature, whereas a new, rationally structured and well planned organization may be very mature in spite of its youth" (Ein-Dor and Segev, 1978, p.1071). In the context of DP, there may be old established DP departments which operate as closed entities, which provide users with only transaction processing applications, and where end users are not involved in systems development. According to Nolan (1979), DP departments having these characteristics are at the earlier stages of DP growth. On the other hand, more mature DP departments

provide users with a broader variety of applications, show more concern with regard to users' need, and call for more user involvement in the development of applications.

Nolan (1979) suggests benchmarks for assisting in the determination of the stage of growth of a DP department [4]. DP expenditure and technology are suggested as first-level benchmarks. Nolan claims that in Stages 1 and 6, the growth rate of the DP budget "tracks" the rate of growth of sales, while it exceeds the growth of sales in Stages 2 and 4, and is lower than the growth of sales in Stages 3 and 5. The technology benchmark helps determine which stage of a given DP is in. For instance, if the rate of growth of DP is lower than the rate of growth of sales and processing is 100% batch, then the firm is in Stage 1, while it is in Stage 6 if technology is more sophisticated, including data base technology, personal computing, etc. However, the reliance on technology to determine the stage of growth of DP can be disputed. Nowadays, even a firm acquiring its first computer is likely to acquire the data base technology, timesharing facilities, and so on.

However, Nolan provides a second set of benchmarks, or more precisely, growth processes that pertain to four aspects of DP: applications portfolio, DP organization, DP planning and control, and user awareness. According to Nolan, the mix of applications evolves from 100 percent operational control systems in Stage 1, to 45 percent in Stage 6, where the portfolio includes 40 percent of management control systems and 15 percent of strategic planning systems. With

regard to the organization of DP, Nolan suggests that DP moves from being a "closed shop" to a situation where "there are layers of responsibility for data processing at appropriate organizational levels"(p.121) in Stage 6. In the early stages (until "sometime" in Stage 3) the focus of DP planning and control activities is internal, and control methods include programming and systems development standards, project management, and so on. In the latter stages of growth, control and planning are more oriented toward outside DP, and include steering committees, data administration, and value-added user charge-back [5]. Finally, Nolan states that in the first two stages of DP growth, users are superficially involved with DP, while they become a "driving force" in the following two stages, being involved in data entry and use of data, and being held accountable for the quality of data and for "value-added end use." In Stages 5 and 6, users and DP "are jointly accountable for data quality and for effective design of value-added applications" (p.121):

The objective of the present study, with regard to maturity of DP, is more to differentiate between more mature and less mature DP departments than to identify the exact stage of growth DP is in. Nevertheless, the four growth processes are a helpful guideline for doing so.

3.2.7 User Attitude Toward UDA

The inclusion of the variable user attitude toward UDA in the present research model was suggested by two sources, namely, the organization change literature and MIS implementation literature.

Beckhard and Harris (1977) suggest the measurement of people attitude toward the change that took place as one approach to evaluate a change effort or program. While the present study does not focus on the change aspect of UDA, the research model includes user attitude toward UDA in order to explore the relationship between this variable and the variable success from a user point of view, that is, user satisfaction.

The inclusion of user attitude in the research model is also suggested by the MIS implementation research. As put by Swanson, "the measurement of user attitudes is now a well-established component of current research in computer-based management information systems" (Swanson, 1982, p.158). For instance, Lucas (1978) found a positive relationship between attitude and MIS implementation success.

The initial research model of the present study suggests that a positive relationship exists between user attitude toward UDA and user satisfaction with UDA.

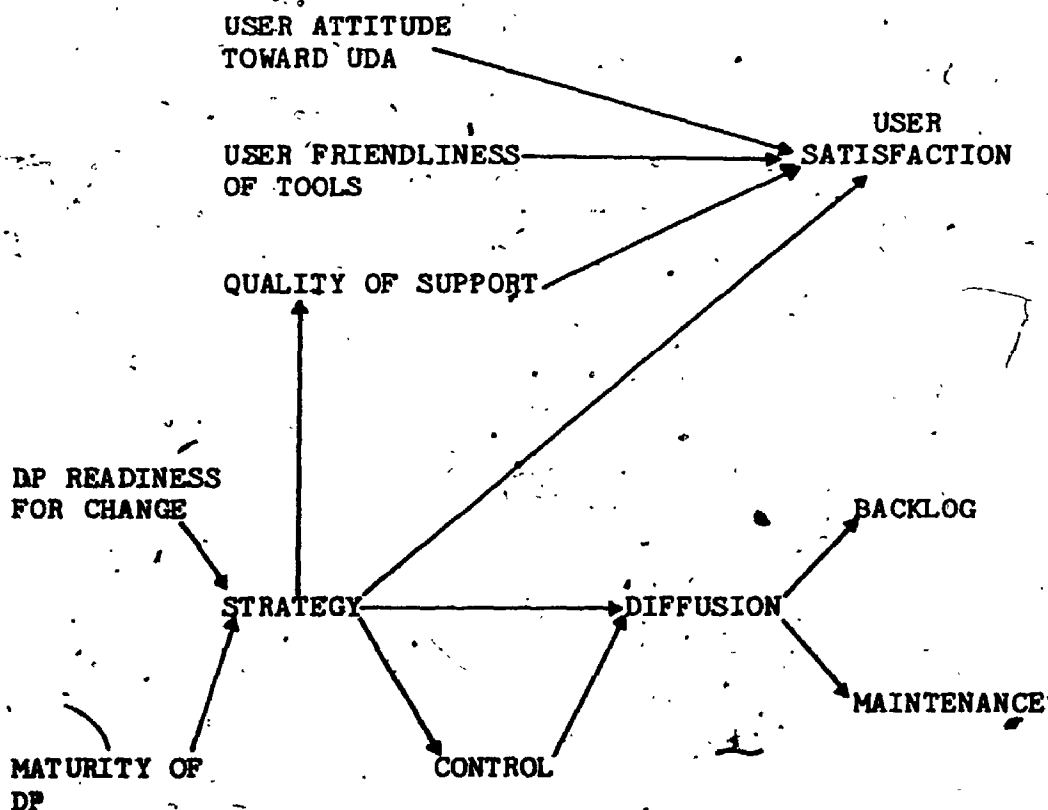
3.3 RELATIONSHIPS AMONG THE VARIABLES

The initial research model is illustrated in Figure 3.1. This section proposes a set of relationships among the variables in the initial model. These propositions will be used to provide direction for the field investigation phase of the study.

Proposition 1: A Push strategy is more likely to take place in firms where DP is more mature. Conversely, Pull strategies are more likely to occur in organizations where the DP department is less mature.

FIGURE 3.1

INITIAL RESEARCH MODEL



According to Nolan(1979) DP departments which are more mature tend to be more perceptive of users' needs. Proposition 1 states that strategies wherein DP is proactive, that is, takes measures to diffuse UDA, are more likely to be found in such organizations than in organizations where there is less concern with user needs.

Proposition 2: A Push strategy is more likely to take place in situations where DP has felt a need for change.

As was previously stated, high backlogs, large maintenance load, and non-traditional requirements are not unusual in many organizations. Yet, they will not always lead DP departments to take measures to promote UDA. Proposition 2 suggests that a DP department is more likely to take such measures when its management has perceived these issues as being important enough that a change must take place in DP's "way of doing business."

Proposition 3A: User satisfaction with the organizational context in which UDA takes place will be greater in the case of a Push strategy than in the case of a Pull strategy.

Proposition 3B: Diffusion of UDA will be greater in the context of a Push strategy than in the context of a Pull strategy.

Proposition 4A: A Push strategy will provide users with higher quality of support than will a Pull strategy.

Proposition 4B: In a Push strategy, the intent of the control mechanisms put in place by DP will be less UDA-restrictive than in a Pull strategy.

These propositions stem from the definitions of the two types of strategies. In a Push strategy it is in the best interest of DP to do

what is necessary to encourage users to adopt the concept of UDA. Propositions 4A and 4B suggest that it will do so by providing users with more support (such as "hot lines," personal consultation, documentation, debugging assistance) in order to make them accept the idea of developing applications. The DP department is also likely to implement control mechanisms which, while providing for data security, data privacy, "wise" use of the computer resources, and so on, will not be intended to prevent UDA from taking place.

Proposition 3B suggests that when DP is proactive with respect to UDA, it will be adopted on a larger basis than otherwise. By the same token Proposition 3A states that when DP is proactive users will be more satisfied with the organizational context wherein UDA takes place.

Proposition 5: The more user friendly the tools are perceived to be by the users, the more the users will be satisfied with the concept of UDA.

Proposition 6: The higher the quality of support users receive from DP, the more they will be satisfied with the organizational context in which UDA takes place.

Proposition 7: There will be a positive relationship between user attitude toward UDA and user satisfaction with UDA.

Proposition 5 states that user satisfaction with the concept of UDA is related to the user friendliness of the tools which are available to him. Proposition 6 suggests that the quality of support users receive will have an impact on their satisfaction with the organizational context in which UDA takes place. Proposition 7 suggests that those users who have a more positive attitude toward UDA

will have higher levels of satisfaction with UDA.

Proposition 8: The less UDA-restrictive the controls established by DP, the greater will be the diffusion of UDA.

Proposition 8 reflects the role of controls in the context of UDA. It suggests that the less the DP department restricts the use of computing by end users, the greater will be the extent to which UDA will be spread throughout the organization.

Proposition 9: The greater the diffusion of UDA, the greater the decrease in DP application backlogs.

Proposition 10: The greater the diffusion of UDA, the greater the decrease in DP perfective maintenance load.

Propositions 9 and 10 describe the relationship between diffusion of UDA and the decrease in the DP backlog and perfective maintenance load. The propositions suggest that the greater the extent to which UDA will be spread in an organization the more the backlog and the perfective maintenance load will decrease.

3.4 SUMMARY

The initial research model described above is the result of an effort to integrate the existing UDA literature to the literature of other relevant areas. The field investigation phase of the study was based on this model, and used the model's propositions as a guide for further exploration of the UDA phenomenon. The next chapter discusses

the research approach chosen to conduct this study and presents the research methodology used.

FOOTNOTES

[1] As discussed in Section 2.1, Lientz et al. define three types of maintenance: (1) corrective maintenance, which consists in emergency fixes and routine debugging, (2) adaptive maintenance, which pertains to the "accomodation of changes to data inputs and files and to hardware and system software" (p.468), and (3) perfective maintenance, which encompasses changes due to user requests for enhancements, improved documentation, and recoding for computational efficiency.

The questionnaire used in the Lientz et al. study is available in Lientz and Swanson (1980). The same questions will be used in the present study in order to differentiate between the categories of maintenance.

[2] The components of user friendliness suggested by Zloof, and listed in Chapter II are repeated here:

1. Minimum concepts are required to get started.
 2. Minimum syntax.
 3. Consistency.
 4. Flexibility.
 5. Not sensitive [to small changes in the query].
 6. Easy to extend and modify.
 7. Minimum exception rule.
 8. Easy detection of errors.
 9. Unified languages.
- (Zloof, 1978, pp.38,39)

[3] It is not meant to be implied here that UDA is impossible when the only technology available is keypunch and punch cards. In fact, such was the technology used in one of the companies studied.

[4] Nolan's stage theory is discussed in Section 2.2.3 and in footnote [7] of Chapter II.

[5] The following, from Nolan (1977), describes what is meant by value-added chargeback.

Data would be purchased from sourcing functional groups as well as from outside sources. In responding to requests from users, Data Processing would provide value-added services by combining, by processing, and by distributing data. The users would be charged for the cost of the data plus the value-added services of processing it. The value-added concept solves a basic problem of current chargeout systems. It provides a quid pro quo for those who bear the costs of collecting data but are not end user benefactors. (Nolan, 1977, p.264)

CHAPTER IV
RESEARCH METHODOLOGY

This chapter discusses the research approach which was adopted to conduct the study and describes the methods used. Section 4.1 first discusses the two objectives of the study, namely, theory generating and theory testing, and presents the rationale behind the choice of multi-phase, multi-method research approach. Section 4.2 describes each of the three phases of the investigation part of the study. The three phases are the identification phase, the interview phase, and the survey phase.

4.1 CHOICE OF AN APPROACH

The purpose of this study is to both develop and provide empirical evidence for a model of the factors of success of UDA. One way of achieving such a purpose would be to develop a model on the basis of generally accepted theories and previous studies, and to test the model by means of a survey research [1]. However, such an approach requires a certain degree of maturity of the field being studied. As Chapter II indicates, up to now, very little research has been done on UDA. The model presented in Chapter III was developed on the basis of "conventional wisdom" with respect to UDA, as well as on inferences from literature not directly pertaining to UDA. The fact that other factors not included in the model could be important in the context of UDA had to be taken into account in choosing a research approach. That is, because UDA is a new phenomenon, it was believed that this study

should put as much emphasis on theory generating as on theory testing.

In the context of the development of middle range theories [2], Merton stresses the importance of the serendipity pattern, that is, "the common experience of observing an unanticipated, anomalous and strategic datum which becomes the occasion for developing a new theory or for extending an existing theory" (Merton, 1958, p.104). Yet, for this experience to be fruitful, the investigator must use a methodology which allows for exploring further in the direction indicated by the anomalous observation.

In order to achieve the two objectives of theory generating and theory testing, this study was conducted in phases and followed a multi-method approach. Ten organizations participated in the research project. In each firm, qualitative research methods including structured and unstructured interviews, study of documents, observation, and comparative analysis were first used. The qualitative approach was adopted in this phase of the study in order to develop a holistic view of UDA and its context in each firm, and to allow for exploration. During this phase, descriptive information on UDA was gathered, the initial research model improved, and the survey questionnaire (which was to be used in the following phase) pre-tested. In the following phase, the questionnaire was sent to users who develop applications in each of the 10 participating companies. This survey phase was aimed at gathering additional descriptive information on UDA from a larger number of users and test the relationships implied by the research model.

The following section provides more details on each phase of the study.

4.2 THE THREE PHASES OF THE STUDY

The investigation effort of this study was divided into three phases. Phase I, the identification phase, involved identifying organizations where UDA existed and obtaining entry in 10 of these organizations. Phase II, the interview phase, was aimed toward developing a "first hand" understanding of UDA and of the various organizational contexts in which it can take place. Second, the field work was a means of submitting the initial research model to the "acid test" of the real world. The purpose of Phase III, the survey phase, was to obtain additional information from a large number of users who develop applications and to test the relationships suggested by the research model, as revised during Phase II.

4.2.1 The Identification Phase

The first task of the investigation part of this study was to identify organizations where UDA existed and to obtain entry in 10 of them. A letter explaining the purpose of the research project, accompanied by a short questionnaire, was sent to the senior DP manager in the 150 largest Canadian companies (as taken from Canadian Business, July 1981). The letter and the questionnaire are reproduced in Appendix K. From the 150 questionnaires sent, 108 responses were received. Table 4.1 outlines the results obtained. Of the 108 firms,

TABLE 4.1
RESULTS OF THE IDENTIFICATION PHASE

	<u>NUMBER OF FIRMS</u>	<u>PERCENTAGE</u>
NO USERS WHO DEVELOP APPLICATIONS	56	52%
LESS THAN 10 USERS	27	25%
BETWEEN 10 AND 50 USERS	11	10%
BETWEEN 50 AND 100 USERS	6	6%
MORE THAN 100 USERS	8	7%
	-----	-----
	108	100%

TABLE 4.2
INDUSTRIAL BREAKDOWN FOR PARTICIPATING FIRMS

<u>FIRMS</u>	<u>INDUSTRY</u>
A, B, G	Financial and Insurance
D, H, J	Manufacturing
E	Forestry
C, F	Utility
I	Communications

only 25 indicated that 10 or more users were developing applications. The 10 firms used in the remainder of the study were chosen from those 25. Table 4.2 lists the industrial breakdown for the participating firms [3]. Eight of the 10 companies had their head office in Ontario, while the other two firms were located in British Columbia.

The main criterion for choosing organizations to be part of the sample was the number of years of experience they had with UDA.

Accordingly, Companies A, B, D, and I were chosen because they had relatively long experience with the Information Centre (IC) approach (2 to 3 years). Companies C, E, and J were chosen because they had a relatively long experience with UDA itself (6 to 12 years), but did not have ICs. Finally, the choice of firms F, G, and H was motivated by the fact that they were in an intermediate situation between the other two groups. While UDA had existed in all three firms for several years, the IC approach was just being implemented in each.

In order to obtain entry in those organizations, either the DP manager or the person responsible for user support was contacted by telephone [4]. The purpose of the research project was explained, as well as the involvement required from participating organizations. Participation in the study implied several hours of interviews with DP management and staff, and with users. It also required someone from the DP department to arrange interviews with users. Finally, DP was asked to provide a list of all users who were known to be or thought to be developing applications in the organization, in order that a survey questionnaire could be sent to those users. In return for their help, the participating organizations would receive a report of the research project.

All ten organizations which were initially chosen to participate in the study agreed to be involved. Before the investigation started each firm was provided with an Executive Summary of the research project and with a list of questions which would be discussed during the interviews. This package is given in Appendix B.

4.2.2 The Interview Phase

What follows describes the techniques used to gather data from both DP departments and from users during the field work stage.

Unstructured and structured interviews took place with DP managers, with people responsible for user-oriented services (such as User Support Group or Information Centre) where such services existed, and with representatives of the user community. The rationale behind the use of both structured and unstructured interviews was twofold. First, it is difficult to stay within the limits of a structured interview. As Alter's experience in his research on Decision Support Systems shows:

Once in the door, the interviews did not proceed according to the initial plan. Although the structured interview involved 1 to 1 1/2 hours worth of information, it was very difficult to go straight through the interview asking questions in sequence. In part, this is because people don't necessarily think about things in the same order. More important was the fact that the respondents weren't captive college sophomores. In addition to explaining the purpose of the study, it often seemed important to build personal rapport with respondents. (Alter, 1975, p.4-15)

Second, and more important in the present context, is the fact that the study is partly aimed at uncovering factors of success for UDA. It was assumed, at the outset of the interview phase, that by their nature unstructured interviews would more easily lend themselves to "discovery" than very structured ones.

Four instruments were used to gather information during the interview phase. First, a questionnaire was developed to collect information from the DP department (Appendix C). However, as in the case of Alter's research, it was very difficult to follow the questionnaire precisely. Moreover, while in some cases the DP manager had the information to answer most of the questions, in most instances the sources of information for the DP questionnaire were not immediately available to the respondent. Finally, as will be discussed further in Chapter V, a series of questions which were aimed at measuring the decrease in the backlog and in the maintenance load, were abandoned 5.

In order to supplement these features of the questionnaire, a check-list was also used in conducting the interviews (Appendix B). The check-list was found to help in insuring that information was obtained from all participating organizations on the key aspects of the study. The check-list also allowed for greater flexibility, both from the part of the respondent and of the interviewer.

The third instrument used was the EDP Profile Questionnaire, developed by McCririck (1979) and discussed in Goldstein and McCririck (1981). This questionnaire was filled out by DP managers themselves and was used mainly to measure the maturity of DP in the organizations studied.

The fourth data collection instrument was a check-list for user interviews (see Appendix D). As was the case with the DP interviews,

the check-list provided enough structure to insure that the major aspects of UDA were covered, while allowing for interviewing flexibility.

More than 150 hours were spent interviewing DP managers and staff as well as users. Sixty two users were interviewed, for approximately one hour each. The remainder of the interview hours were devoted to DP management and support staff. In addition to the information provided during the interviews, some participants (both users and DP people) provided internal documents which in most cases were found to be useful in better understanding UDA and its context in the firms studied.

Once the interview data were gathered from one organization, efforts were made to analyze them right away. As Glaser and Strauss (1970) note, in research aimed toward the discovery of theory, it is crucial that data gathering and data analysis go on simultaneously. If this procedure is followed, the investigator is less likely to find himself overwhelmed by a mass of data out of which he has to make sense. In the present study, data gathered in one organization was compared to data already collected from others, so as to determine similarities and differences. In doing so, the research model evolved as the investigation took place. That is, it happened that some variables in the model were eliminated because they were not found important in any of the organizations studied up to that point. On the other hand, new variables entered because they were found to be important in a large number of instances.

Following Phase II, a separate scenario was written to describe UDA in each of the firms studied (Appendix F). The other major output of Phase II was an improved version of the model of factors of success for UDA. Chapter V discusses the results of Phase II in detail; the scenarios are summarized, the variables of the initial research model are examined with regard to their relevance in explaining success of UDA, and the revised version of the model is presented.

4.2.3 The Survey Phase

This section summarizes the purpose and the research methodology for the survey phase of the study. Chapter VI will discuss them in more details, along with the results of this phase.

The purpose of the survey phase was to obtain additional descriptive information from a large number of users, and to test the relationships implied by the research model, as revised in Phase II.

A survey questionnaire was sent to a total of 1074 users. The questionnaire, copy of which is in Appendix E, was pre-tested during the interview phase of the study. Of the 1074 questionnaires sent, 429 (40%) were returned, and 272 (25%) were actually used in the statistical analyses. Table 4.3 summarizes the sources of the 15% difference between the number of questionnaires returned and used. The reasons for this difference are discussed further in Section 6.1.

TABLE 4.3
RESPONSE RATES

	NUMBER	PERCENTAGE
TOTAL NUMBER OF QUESTIONNAIRES SENT:	1074	100%
TOTAL NUMBER OF QUESTIONNAIRES RETURNED:	429	40%
Non-usable questionnaires		
Users who do not develop applications:	83	
DP professional working in DP:	18	
Incomplete:	15	
Engineering applications:	3	
Returns to sender:	8	
Usable but received after cutoff date:	30	
TOTAL:	157	15%
TOTAL NUMBER OF QUESTIONNAIRES USED:	272	25%

A test for non-response bias was performed and no significant differences were found between respondents and non-respondents.

4.3 SUMMARY

This chapter presented the research methodology that was used in the study. The methodology involved three phases: the identification phase, the interview phase, and the survey phase. The data gathered in the latter two phases will be analyzed in the next two chapters: the interview data in Chapter V, and the survey data in Chapter VI.

FOOTNOTES

[1] Lucas' (1976) study on the implementation of Information Systems is an example of such an approach.

[2] As defined by Merton, theories of the middle range are "theories intermediate to the minor working hypotheses evolved in abundance during the day by day routines of research, and the all-inclusive speculations comprising a master conceptual scheme from which it is hoped to derive a very large number of empirically observed uniformities of social behavior" (Merton, 1958, pp.6-7). Other authors, such as Simon (1978) call this type of theory a model, that is, "a set of propositions relevant to one corner of a field or to a few related phenomena ..." (Simon, 1978, p.64).

[3] Since some of the participating organizations expressed the desire that their names be held confidential, all the participating firms are labeled with a capital letter only.

[4] The questionnaire in Appendix A requested the name of a person in the firm who could be contacted in order to obtain more information on UDA.

[5] The questions pertaining to diffusion of UDA, decrease of DP application backlog, and decrease of the perfective maintenance load are QUESTION B1, QUESTION B3, QUESTION B4, and QUESTION C4 to QUESTION C12.

CHAPTER V

PRESENTATION AND ANALYSIS OF THE INTERVIEW DATA

This chapter presents the results of Phase II of this study, that is, the interview phase. Interviews conducted with DP managers, with DP professionals who are in charge of supporting users, and with users themselves, were the primary source of data for this phase. Secondary sources of data were: a DP profile questionnaire, internal documents that were made available either by DP or by users, and observation. Phase II had two major purposes. First, it was aimed toward getting a "first hand" understanding of UDA and of the various organizational contexts in which it can take place. Second, the field work was a means of submitting the initial research model to the "acid test" of the real world. This test included two major questions: (1) Is there empirical evidence to support the factors of success and the relationships included in the initial model? and (2) Does the experience the firms in the sample have had with UDA suggest the inclusion of other factors in the model? The major outcome of Phase II is a revised version of the model of the factors of success of UDA.

The organization² of the chapter reflects the purposes of Phase II. Section 5.1 describes the study settings, the modus operandi of UDA in the organizations visited, the programming tools, and so on. Section 5.2 discusses the appropriateness of the initial research model in explaining success of UDA. Finally, Section 5.3 proposes a revised version of the model [1].

5.1 BACKGROUND INFORMATION ON THE FIRMS STUDIED

This section describes the settings wherein the field work took place. The inclusion of this section serves two purposes. First, it is a means of defining the domain of the present study. As suggested by Dubin:

Every theoretical model, if it is complete, must specify the boundary within which the units interact lawfully. Beyond that boundary it may not be at all clear that the units will continue to interact by the specified law; or that all the units will remain in the system; or even if the system will remain intact. (Dubin, 1971, p.59)

The second purpose served by the inclusion of descriptive information is to provide the reader with background information on the context of UDA in the organizations studied. This background information constitutes the basis for the discussion which takes place in the remaining sections of the chapter.

Tables 5.1 through 5.3 outline the major dimensions along which information was gathered in the enterprises studied. Appendix F provides a complete scenario of UDA for each of the 10 firms. For the remainder of this section, the reader is strongly encouraged to refer to these Tables; the text which follows comments heavily on their contents.

5.1.1 The Companies

Refer to Table 5.1. The ten firms in the sample are among the 100 largest companies in Canada, according to Canadian Business, July 1981. Two of the firms (A and G) are financial institutions, one (B) is an insurance company [2]. Two of the firms are utilities (C and F), one (I) is in the communications industry; these three firms have their prices set by regulatory boards. D, H, and J are manufacturing firms, and E is in the forestry industry. The assets of the companies studied vary from \$900 million to more than \$60 billion, their sales, where applicable, vary from \$800 million to more than \$8 billion, while the number of employees range from 2,400 to nearly 40,000. Half of the organizations studied (C, D, H, I, J) are subsidiaries of a parent organization.

5.1.2 Overview of DP

Refer to Table 5.2. All of the organizations studied have been using computers for commercial DP for more than 15 years. As indicated by Table 5.2, the 10 firms are primarily "IBM shops." The monthly rental-equivalent for hardware and communications costs varies from \$32,000 to more than \$5 million, and the number of DP employees varies from 55 to approximately 1,500. All the companies but one (F) have implemented data base technology.

In eight of the companies, the senior DP executive reports to a Vice-president or to a Senior Vice-president, while in the two

TABLE 5.1
THE COMPANIES

	A	B	C	D	E	F	G	H	I	J
INDUSTRY	FINANCIAL	INSURANCE	UTILITY	MANUFACTURING	FORESTRY	UTILITY	FINANCIAL	MANUFACTURING	COMMUNICATIONS	MANUFACTURING
PRICES SET BY A REGULATORY BOARD?	NO	NO	YES	NO	NO	YES	NO	NO	YES	NO
ASSETS	>\$50 Billion	>\$4 Billion	\$1.3 Billion	\$2.8 Billion	\$1.05 Billion	\$1 Billion	>\$60 Billion	\$1.7 Billion	\$2.3 Billion	\$900 Million
SALES	NOT APPLICABLE	>\$4 Billion	\$1.1 Billion	>\$8 Billion	NOT PROVIDED	\$800 Million	NOT APPLICABLE	\$1.4 Billion	\$900 Million	\$800 Million
NUMBER OF EMPLOYEES	23 000	4 500	2 700	39 000	7 000	2 400	28 000	11 500	14 400	5 000
SUBSIDIARY OF A PARENT ORGANIZATION?	NO	NO	YES	YES	NO	NO	NO	YES	YES	YES

financial institutions (A and G) the senior DP executive reports to the President. DP responsibilities are centralized in most of the firms studied. D and H (both manufacturing firms) have formal systems groups based in user areas; in these companies, the groups were created by DP, which transferred analysts to user departments. In both cases, the systems analysts have, as a formal responsibility, performing systems analysis and design tasks for new applications, as well as transmitting to DP users' requests for changes in existing systems. At firm I, systems groups in user areas exist as well, but most were not purposively created by DP. Rather, they evolved around a nucleus of interested users, most of whom had a computer background. At J and E, there are divisional systems coordinators who are responsible for the interface between the corporate DP department and the users in certain divisions.

Users are charged for systems development in all the firms except for F, H, and I, and are charged for their use of computer resources in all firms except A, F, H, and I. However, at I, users who use computer resources in the context of UDA are charged for their use of the computer.

F and J do not have a DP steering committee while the other firms have one. In general, the role of the committee is to approve priorities established by DP, and to approve major projects or acquisitions.

TABLE 5.2
OVERVIEW OF DP IN THE FIRMS STUDIED

	A	B	C	D4	E	F	G	H	I	J
NUMBER OF YEARS HAS BEEN USING COMPUTERS	20	20	15	20	18	20	20	25	20	20
HARDWARE	3: IBM 3033 1: IBM 3031 1: IBM 370-148 1: AMDAHL V7	IBM 3081	IBM 3031 AMDAHL V6	IBM 3081 AMDAHL V7	IBM 4341	IBM 3032 IBM 148	2: AMDAHL V8 AMDAHL V7 AMDAHL V8 IBM 3081	IBM 3081 IBM 4341	DEC 10 AMDAHL V8 IBM 3081 IBM 4341 DEC 10 DEC 10	IBM 4341 IBM 370-148
MONTHLY HARDWARE & COMMUNICATIONS RENTAL EQUIVALENT	>\$2.5 million	>\$200 000	\$225 000	\$800 000	\$32 000	\$200 000	\$5 million	\$250 000	\$460 000	\$200 000
NUMBER OF DP EMPLOYEES	1 500	160	130	300	55	120	1 200	200	390	70
DATA BASE TECHNOLOGY USED	YES IMS	YES IMS	YES ADABAS	YES IMS	YES (Brand new) IDMS	NO	YES IDMS	YES IMS	YES IMS	YES IDMS
TO WHOM DOES THE SENIOR DP REPORT?	PRESIDENT & COO	VP-FINANCE	SENIOR DP ACCOUNTING & REGULATION		VP-CORPORATE SERVICES YES (SOME DIVISION COORDINATORS)	VP-PERSONNEL & SYSTEMS	PRESIDENT	VP-ADMINISTRATION	VP-ADMINISTRATION	VP & TREASURER YES (SOME DIVISION COORDINATORS)
FORMAL SYSTEMS ARE USERS CHARGED FOR SYSTEM DEVELOPMENT?	YES	YES	YES	YES	YES	NO	YES	NO	NO	YES
ARE USERS CHANGED FOR OTHER DP RESOURCES?	NO	YES	YES	YES	YES	NO	YES	NO	NO	YES
STEERING COMMITTEE?	YES	YES	YES	YES	YES	NO	YES	YES	YES	NO
FORMAL DEVELOPMENT METHODOLOGY?	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES
PROJECT MANAGEMENT METHODOLOGY?	YES	YES	YES	YES	YES	YES	YES	YES	YES	YES
LARGE BACKLOG?	YES	YES	NO	YES	NO	YES	YES	YES	YES	NO

All the DP departments make use of formal methodologies for systems development and programming. DP managers at C, E, and J state that the backlog of application development projects is low, while the seven other managers estimate the backlog as being several years long.

5.1.3 The DP Side of UDA

Refer to Table 5.3. In all the firms studied, DP provides some assistance to users who develop applications. Among the 10 firms two modes of support were identified: seven of the firms had implemented the Information Centre (IC) concept [3], while in the remaining three (C, E, and J) assistance is provided to users by "product coordinators."

Organizational Location of UDA Support

In seven of the 10 companies, responsibility for supporting users in the context of UDA ultimately belongs to the Systems Development group. Four of the seven ICs (A, G, H, I) lie organizationally within the Systems Development group. In the three firms where UDA support is provided by product coordinators, these product coordinators are members of Systems Development. The three companies which are exception are: B, where the IC is under the responsibility of Office Systems, D where the IC supervisor ultimately reports to the manager of Technical Development, and finally, F, where the activities of the IC are under the responsibility of the Systems Services manager.

TABLE 5.3
DP SIDE OF UDA

	A	B	C	D	E	F	G	H	I	J
HOW IS UDA FACILITATED?	IC	IC	-SMALL SYSTEMS GROUP; MODELER -COORDINATOR; EASYTRIEVE	IC	COORDINATOR	IC	USER COMPUTING FACILITIES	TIMESHARING SUPPORT GROUP	IC	COORDINATORS
WHERE DOES "FACILITATOR FIT IN DP"?	SYSTEMS DEVELOPMENT	OFFICE SYSTEMS	SYSTEMS DEVELOPMENT	TECHNICAL DEVELOPMENT	SYSTEMS DEVELOPMENT	SYSTEMS SERVICES	SYSTEMS DEVELOPMENT	SYSTEMS DEVELOPMENT	SYSTEMS DEVELOPMENT	SYSTEMS DEVELOPMENT
LENGTH OF TIME SINCE UDA HAS STARTED TO BE FACILITATED AS IT IS NOW	3 1/2 YEARS	1 1/2 YEAR	MODEL: 9 YEARS EASYTRIEVE: 6 YRS	2 YEARS	10 YEARS	9 MONTHS	2 YEARS	11 MONTHS	4 YEARS	ASI-ST: 10 YRS IFPS: 6 MONTHS
NUMBER OF REGISTERED USERS	170	150	70	190	70	90	100	150	600-700	ASI-ST: 75 IFPS: 20
WHY INTRODUCE SUPPORT TO UDA?	LARGE BACKLOG OF SMALL APPLICATIONS	OFFICE AUTOMATION INCREASE PRODUCTIVITY	MODEL: EVOLVED EASYTRIEVE: BACKLOG	BACKLOG AND TO MAKE DATA AVAILABLE	INCREASE PRODUCTIVITY	BACKLOG	-ACCESS TO DATA -OUTSIDE TIME-SHARING -BACKLOG	OUTSIDE TIMESHARING	OUTSIDE TIMESHARING	BACKLOG OF SMALL APPLICATIONS
TOOLS AVAILABLE	GIS, SAS, ADRS, APL, IFPS, BASIC	GIS, SAS, SAS/GRAPH, APL/DI, APL, ADRS, GDDM	MODELER, EASYTRIEVE	* ADRS, PLANCODE, GIS	FORESIGHT	PLANCODE, APL, ADRS	FOCUS, ADRS, IFPS, APL, APL/DI, EASYTRIEVE	EASYTRIEVE, PLANCODE, ADRS, GDDM	FORTRAN, COBOL, PASCAL, EASYTRIEVE, SAS, SAS/GRAPH	ASI-ST, IFPS
LOCATION OF USERS	MOST ON SAME SITE AS IC	ALL ON SAME SITE AS IC	MOST ON SAME SITE	MOST REMOTE FROM IC	MOST REMOTE FROM DP	ALL ON SAME SITE AS IC	MAJORITY ON SAME SITE SOME REMOTE	REMOTE (SAME CITY)	REMOTE	HALF ON SITE; HALF REMOTE
SEPARATE LOCATION FOR CONSULTANTS?	YES	NO	NO	YES	NO	NO	NO	NO	NO	NO
WALK-IN CENTRE?	YES	YES	NO	NO	NO	YES	YES	YES	NO	NO
NUMBER OF STAFF	6+1 MGR	5	MODELER: 3+1 MGR (PART-T.) EASYTRIEVE: 1	3+1 MGR	1	3	6+1 MGR	6	5+1 MGR	1: ASI-ST 3: IFPS
PROMOTE SERVICES?	NO LONGER	YES	MODELER: YMS DP: NO	YES BUT LESS THAN BEFORE	NO	YES	YES	YES TO OUTSIDE TIME-SHARING USERS	YES	NO

While at A, D, G, and I, there exists an IC manager or supervisor, it is not so at B, F, and H. In these three firms, the IC staff reports directly to a manager who has other responsibilities as well: Technical Transfer and Office Automation at B, IS Development Services at F (such as support of the project development methodology and of the project management methodology), and Data Base Administration at H.

Basic Facts

The "length of time since UDA has started to be facilitated in the present fashion" reported in Table 5.3 calls for some explanation. For the seven firms where an IC or a formal support group exists, the number of years of experience with UDA reported is the number of years since the IC implementation. In the case of C, E, and J, the number of years of experience reported pertains to the number of years since the introduction of a particular software tool in the organization. This clarification is necessary because for most of the seven firms where the IC concept has been implemented, UDA is not completely new. At B for instance, actuaries had programmed in FORTRAN for several years; at F, DP had made MARK IV available to certain users for at least six years; at D, systems analysts located in user departments had used EXTRACTO for several years; company I had had a timesharing support group, and so on.

The number of registered users indicated in Table 5.3 cannot be automatically interpreted as the number of users who actually develop their own applications. In the cases of A, B, D, F, G, and H, the figure represents the number of users who have an account number in

order to use the computer. For these six companies, the source of this information is a list of registered users. In the case of company I, there are approximately 1200 account numbers, but according to a senior IC consultant, there are between 600 and 700 individual registered users. At C, E, and J, the number of users is an estimate provided by each product coordinator. However, these product coordinators felt that the estimate they gave was fairly accurate.

In none of the 10 organizations studied does the number of registered users represent the number of users who actually develop applications. A user may very well have an account number but only run applications developed by other users, or just enter data, or, as in the case of B, use a text editor. At I, for instance, it was believed that out of the 600 to 700 registered users, approximately 450 developed applications, while at H, it was estimated slightly more than one third or so of the timesharing users actually developed applications.

Reason for Introducing UDA

A large backlog of DP projects, a significant portion of which consisted of small, one shot types of applications, is the single most often cited reason for introducing the IC concept (A, C for EASYTRIEVE, D, F, J). Reducing the usage of outside timesharing is another reason for introducing the IC concept (G, H, I). At E, the introduction of FORESIGHT, a planning tool, was motivated by the desire to increase the productivity of accountants who, at budget time, spent several weeks preparing budgets, using pencil and paper methods (for more details see

Scenario E, in Appendix F). At B, while the backlog of applications is mentioned as one motivator, the IC is the first step of a larger effort toward Office Automation, the overall objective of which is to increase productivity. To summarize, three major reasons were proposed by DP managers to explain the introduction of UDA in their firms: to cope with the large backlog of small application requests, to repatriate outside timesharing usage, and to increase people productivity.

Tools Available

There is much commonality among the firms with regard to UDA software tools offered and supported by DP. Table 5.3 lists the tools used by each firm, while Table 5.4 displays the firms using each tool, and also indicates the tasks which can be performed using each [4]. Appendix G provides a brief description of each tool.

It should be noted that the tools listed in Table 5.4 are the tools made available by DP through an IC or a group with similar responsibilities, or supported by a coordinator. In most of the firms, other tools not generally recognized as UDA tools, are also used by some users. Examples include FORTRAN programming at B, C, D, and H, use of MARK IV at E, and F. Users who use such tools seem to have received special treatment, because of their particular needs. In addition to these in-house tools, in several of the firms there are users who use tools provided by outside timesharing firms. Those include RATEMOD at C, GE.FALLII at H, etc. Most of the time, these tools serve very special needs which cannot be satisfied by the tools provided in-house. Other times, they were used before other tools were

TABLE 5.4
TOOLS IN THE FIRMS STUDIED

<u>TOOL</u>	<u>FIRMS OFFERING TOOL</u>	<u>QUERY</u>	<u>REPORT GENERATOR</u>	<u>GRAPHICS</u>	<u>MODELING LANGUAGE</u>	<u>GENERAL PURPOSE PROGRAMMING LANGUAGE</u>
ASI-ST	J	X	X			
GIS	A,B,D	X	X			
EASYTRIEVE	C,G,H,I	X	X			X
APL/DI	B,G	X	X			X
FOCUS	G	X	X	X		X
ADRS-II	A,B,D,F, G,H	X	X			X
SAS/GRAPH	B,I			X		
GDDM	B,H			X		
APL	A,B,D, F,G					X
BASIC	A					X
FORTRAN COBOL PASCAL	I					X
IFPS	A,G,J				X	
PLANCODE	D,E,H				X	
MODELER	C				X	
SAS	A,B,I					(STATISTICAL ANALYSIS PACKAGE)

made available in-house, and the users were reluctant to convert their applications [5].

Physical Location of UDA Support

As was mentioned in Section 2.2.2., an IC is generally defined not only as an organizational entity but as a physical location. According to IBM's philosophy, the physical "outlook" of the IC should be the following:

There needs to be space set aside exclusively for the Information Centre and this space should be centrally located (central to users, that is); it should be large enough that several users can work in privacy; there should be a debug desk where paper can be spread out;...There should also be a terminal facility for the Information Centre staff and for users who don't have terminals of their own. The Centre might provide additional "walk-in" terminals. (Grindlay, 1980, p.85)

Of the 10 firms studied, only the ICs at A and at B had all the features recommended by IBM. And, still, at B, the walk-in terminal centre had been installed only six weeks before this study's interviews took place. All the other firms differ from IBM's recommendation on at least one characteristic. Companies C, F, G, H, and I, while they say they have implemented the IC concept, do not provide space set aside exclusively for the IC. In these firms, the IC consultants are more or less dispersed throughout their DP group. At F for instance, all DP staff is located on the same floor, and there is no special indication of who is an IC consultant. At H and I, while the IC consultants have their desks grouped together, they are not actually separated from other DP professionals. At G, the User Computing Facilities

consultants are dispersed among people from the larger DP group to which they belong (called Interactive Computing Facilities). At C, E, and J, the situation is the same: product coordinators are located within their DP group.

Some of the firms provide a walk-in terminal centre especially for UDA (A, B, F, G, H), others have pools of terminals that are for more general use (C, E) while others do not provide such facilities (I, J). D is a special case where the IC has four terminals which are mainly used for training. While under special circumstances, users can use these terminals, the IC strongly encourages them to make sure that a terminal is available near their work area. If a terminal is not available users are advised to get one of their own before receiving training with a tool.

For half of the companies studied a large portion of the user population is located far from those who provide support. This is the case for D, E, H, I, and J. At D for instance, the IC is located at the company's headquarters. However, the headquarters consists of a large complex of buildings, so even if the majority of users is located at the headquarters, a user may still be a few miles from the IC. Furthermore, several users are located in other cities, often hundreds of miles from the IC. The cases of E, I, and J are quite similar. At H, most users supported by the Timesharing Support Group are at different locations than the support group's location. However, all users are located in the same city.

Support Provided

The size of the ICs, in terms of number of consultants, is quite similar among the 7 firms which have introduced the concept. Most of them have 5 to 6 consultants. There are two firms where the number of consultants is lower, namely D and F. At F, the IC started its activities 9 months ago. Having a small number of consultants in the early stages seems to be common for many ICs. At G for instance, one year ago the User Computing Facility had two consultants and one manager. A's IC started with two consultants. In the case of D, where the IC has been existing for two years, the number of consultants is also three. According to the IC Supervisor, this number is too low to allow the IC staff to provide all the support they should provide. However, for the time being, the budget devoted to the IC does not allow for any expansion [6]. In the cases of the organizations which do not have an IC, there is generally one coordinator per tool. However, at C, support of MODELER (name disguised) is available from all members of the Small Systems Group [7]. Yet, MODELER support is a part-time activity for the group, whose main responsibility is to develop small systems for the user community.

Some components of support provided to users are common among the 10 firms studied. These common components are (1) training users with the tools supported, (2) providing debug assistance, (3) providing technical support and (4) performing administrative tasks such as providing new users with identification codes, scheduling training sessions, maintaining usage statistics, and so on.

In general, training is provided through seminars which take place on a regular basis, or on demand when there is a sufficient number of users to form a class. Training with EASYTRIEVE, in all the firms where it is available (i.e. C, G, H, I), includes a audio-visual course and a "hands-on session" following the course. At B, user training with all the tools is on a "one-on-one" basis, on demand.

The purpose of "Technical Support", as defined by the brochure prepared by the IC staff at D, is "to make the technical side of demand processing UDA as transparent as possible." Technical Support includes setting up files or copies of production files for users, advising users when they want to purchase hardware, "negotiating" service contracts with the computer operations department, and so on.

In addition to the three basic components of user support, IC consultants and product coordinators are responsible for background support activities. Examples of such tasks are the following. At A, the IC staff is responsible for helping in the preparation of a business case for those users (or user department) who want a terminal of their own. At B, D, and I, IC consultants are formally responsible for assisting users in planning their requirements for computing capacity. At the time the study took place, some consultants at F and at G were heavily involved in evaluating new programming tools.

Promotion of the services provided to users or of the programming tools made available to the user community is another background support activity of IC consultants or of product coordinators.

However, it should be noted that in the three firms where there is no IC or formal support group (C, E, J), there is very little promotion done. The only instance is at C, where the Small Systems Group manager, who created MODELER, promotes the tool in the user community. In the cases of the firms where an IC exists, except for A and D, the IC services are promoted. At A, the IC has entered a stage where the IC management has a mandate for controlling usage for the IC, and for putting more emphasis on consolidating the services already provided to the user community rather than increasing the number of users. At D, the IC staff still makes some presentations to the user community; however, promotion activities are quite reduced compared to what they were in the early days of the IC. According to the IC supervisor, this reduction in promotion activities is due to limited IC staff resources, and to current limitations in computing capacity at D. At H, while the services of the Timesharing Support Group are promoted in the user community, promotion efforts are focused on the "large outside timesharing users," in order to motivate these users to use the in-house timesharing services.

In the majority of the firms studied, an implicit rule of IC consultants and product coordinators is to never write code for users. Experience has shown that if they do write code for users, the consultants will also be responsible for making changes to the code when necessary. Consequently, the IC or the product coordinator would become a parallel DP department. Such a consequence is contrary to the philosophy of UDA in most of the firms that were in the study sample. This philosophy calls for helping users to become as "independent" as

possible.

However, as for most rules, this rule of "no code writing" has its exceptions. At H, for instance, the IC consultants state that: "sometimes when you have tried everything else, as a last resort you will write a few lines of code." In company B, by top management decision, the IC is also responsible for developing some special applications for users. According to the IC manager at B, these applications are not very successful. When changes in their business needs require changes in an application, users are reluctant to do the changes themselves, and the IC consultants are not always available for making such changes. In the opinion of the IC manager, users for whom the IC develops applications have a lower level of satisfaction with their applications and with the IC services than those users who do all the work on their own. At C, those who provide support for MODELER claim that writing a few lines of code for users helps show them how easy programming is. Moreover, application development for users is a formal part of the responsibilities of the user support personnel at C. Support of MODELER is provided by the Small Systems Group whose main responsibility is to develop small applications. EASYTRIEVE support is provided at C by a product consultant who also develops some EASYTRIEVE applications for the user community.

5.1.4 The User Side of UDA

The following present some results of the interviews conducted with users in the firms studied, and focuses on three aspects of the "user side of UDA." Those aspects are (1) a taxonomy of users, (2) computer background of users, and (3) a taxonomy of user developed applications.

A Taxonomy of Users

A total of more than 60 users were interviewed in the 10 organization studied. Among other things, users were asked what their responsibilities were in their department, if they had been developing applications for long, if they developed applications for themselves or for other people, and so on (the checklist used for user interviews is in Appendix D). After interviews had been conducted in three or four organizations, patterns started to emerge, with regard to the reasons for which a user would develop applications, for whom he developed applications, etc. At this stage it appeared appropriate to develop a taxonomy of users. The taxonomy is not only useful in helping better describe the user population; it also suggests that some problems or benefits of UDA are different from one category of users to the next.

The users interviewed were found to belong to one of the following categories: (1) Micro DP Department, (2) Staff Analysts, and (3) Opportunity Seekers. Table 5.5 indicates the percentage of the users interviewed falling in each category.

TABLE 5.5
 PROPORTION OF INTERVIEWEES PER TAXONOMY CATEGORY

	<u>NUMBER</u>	<u>PERCENTAGE</u>
MICRO DP DEPARTMENT	27	42.2%
STAFF ANALYST	20	31.2%
OPPORTUNITY SEEKER	17	26.6%
	-----	-----
	64	100.0%

Micro DP Department. The Micro DP Department category consists of users who respond to requests for applications from other users in their department (and sometimes from other departments). For most of these users, developing applications for others is a formal duty. This category was labeled Micro DP Department because their task is very similar, although it is on a much smaller scale, to the DP function. Moreover, they sometimes have to face problems similar to DP's. For instance, some say that users do not always know what they want, and that once an application is developed, many users will request changes. Also, these user-developers are not always able to respond to requests immediately. Hence, backlogs exist in these Micro DP Departments. The types of applications they develop vary from merely extracting data from production files or data bases to full-transaction processing applications in some cases. The latter type of applications involve the creation of local files and the update of these files. Some applications are "one-shot", but all the Micro DP Department users also said that they had developed applications which are run on a regular

basis. Micro DP Department users generally run the applications themselves.

Most users in the Micro DP Department category at D, H, and I had a different status from those in other firms; that is, they were generally part of a systems group in a user department. In other organizations, a department manager or supervisor might choose from among his staff one individual who is interested in learning how to use programming tools, have that person trained with a tool, and make him responsible for responding for requests from other users in the department. The department may also hire somebody especially for performing that job. In such a case, the tendency is to hire either a programmer, or at least somebody who has some background with programming.

However, at D, H, and I the situation is somewhat different. As indicated previously, these three organizations have formal systems groups within user departments. Typically, the role of such groups is to work in conjunction with DP in the development of production systems.

The question might be raised as to why these "users," who sound more like DP professionals, are included in the Micro DP Department category, or even why they are considered as users? The reason for including them in the user population is that both themselves and DP consider them as end users. They are included in the Micro DP Department category because their task in the context of UDA is

essentially the same as the task of the other members of the category. While they may have a stronger computer background than other users in the same category, it is not necessarily so. Interviews took place with some systems analysts from groups based in user departments who had much less knowledge about programming than, say, a "business analyst" with a computer science background.

Staff Analyst. The second category of users is called Staff Analyst. Most of these users have as a main task to perform studies, the results of which are to be used by management to control, plan, or make decisions. These users have titles such as business analyst, financial analyst, business planner. In general, these users do not develop applications for their personal use, but neither do they respond to precise requests from other users, as is the case for the Micro DP Department category. The applications the Staff Analysts develop are tools they use to perform the studies for which they are responsible. The idea of developing applications is often suggested by their supervisor. They may spend a large amount of their time in developing applications but they see it as a means to an end, not an end in itself.

Opportunity Seekers. The third category, Opportunity Seekers, consists of individuals who have much expertise in their own function, who often are at a managerial level, such as supervisor or budget director, but who typically belong to a staff department of the organization. These users have some similarities with users in the Staff Analyst group, since they also are specialists. However, a major

difference from the former category comes from the fact that the Opportunity Seekers appear to be more independent. They also seem to be more creative and proactive, in the sense that they generally do not wait until somebody asks for a new application, but rather actively identify problems which could be solved, or at least analyzed, using the computer. The applications they develop vary from data extraction, to models, to applications which create files, update them and produce reports used by departments they serve or by top management.

According to the above classification of users, we might say that at least for the 10 organizations studied, end users are still quite far from a situation where "everybody drives his own car rather than obtaining the services of a professional driver" (Sammet, 1969, p.730). To pursue the analogy further we could compare today's DP professionals to car racers, and most of the users who develop applications to chauffeurs.

Computer Background of Users

The computer background of the users interviewed was most varied. Some users had absolutely no knowledge of computer programming before starting to develop applications, while some had a computer science background and had worked for several years as DP professionals. Several users had acquired their knowledge of programming by actually "trying out things" rather than by formal courses.

The interviews suggest that users with different backgrounds put different types of demands on DP and on the support staff. For

instance, when complaints were made about the lack of knowledge of the programming tools by the IC consultants, they were likely to come from somebody who had quite a good computer background. On the other hand, users who mentioned the importance of "people orientation" of the consultants typically were individuals with relatively little computer background.

From the interviews, it appears that computer background has an effect on the perception of the ease of use of a tool. With respect to a given tool, some users would say that they had much difficulty learning how to use it while others would not even think of learning to use the tool as an issue. Users with little computer background also tend to ask consultants' assistance more often than those who have a stronger background.

A Taxonomy of User Developed Applications

One purpose of the interviews with users was to find out what type of applications users develop. The literature on UDA suggests that users are more likely to develop one-shot applications, where data are extracted, "massaged," and a report produced. As previously mentioned in this chapter, in several of the organizations studied, programming tools and user support were introduced to allow users to develop such applications.

However, the type of applications users actually develop is much more varied. As reported in Section 2.3, Rockart and Flannery (1981) identify six categories of applications APEX users developed. These

categories are: (1) APL prototypes, (2) APL routinized batch systems, (3) General-Purpose Models and Statistical Routines, (4) Report Reformatters, (5) Inquiry and Status Reporting, and (6) Text-Oriented Applications. However, these categories are not readily usable for the applications developed by the users interviewed in the present study. First of all, text-oriented applications, such as the use of electronic mail and of text editing packages were excluded from this study. Second, the category APL prototypes is likely to encompass applications with several different purposes. Calling them "prototypes" merely indicates that these applications will eventually be rewritten in a more efficient language (e.g. COBOL). Moreover, Rockart and Flannery report that at APEX, "the word 'prototype' has become a buzzword which is often used to mask a quick and dirty system that ends up lasting years" (Rockart and Flannery, p.357). The applications developed by the users in the present study were found to fall in six categories, which are described below.

Models. Several users indicated that they were developing models, the parameters of which were changed to answer "what if" questions. Such models were used to evaluate different investment opportunities, support buy or lease decisions, prepare for regulatory board hearings (in the cases of C, F, and I), and so on. A large proportion of the applications developed by Staff Analysts falls in this category. The most common tools used to develop such models are IFPS, PLANCODE, and APL.

Transaction Processing Applications. The second category of applications is called transaction processing applications, and is very similar to the APL routinized batch systems described by Rockart and Flannery. These applications are labeled transaction processing because they update files which are, in general, created by the user. That users develop such applications was a concern to some of the DP managers in the organizations studied (at A and at I for instance). Such applications were more common in some firms than in others. This is partly explained by the type of tools available to users; for instance, very few such applications were found at J. However, in this firm the only tools made available to users by DP were IFPS and ASI-ST. The latter does not allow users to create files, while the purpose of the former is not to develop transaction processing applications. Yet, as will be discussed later in this section, some users "stretch" the features of a tool and use it for purposes quite different from the tool's original purpose. The types of tools available is not the only explanation for finding different proportions of transaction processing applications between firms. At H for instance, most of the applications developed by users have a transaction processing "flavor." This may be explained by the fact that, as indicated by the DP manager, there is still a large proportion of the production systems which remain to be developed. As said by several of the users interviewed at H: "Everything is still done manually here. If we wait for DP to develop the applications we need, we will never get them. So we do it ourselves."

Data Analysis Applications. The third category consists of applications which perform data analysis, generally using statistical packages. Relatively few of these applications were identified and they were mainly developed by those users who were in marketing and economic studies areas. SAS was the tool most often used for such applications.

Queries. The fourth category consists of applications which extract data from a file or a data base and produce a list or report, without performing any calculations on the data. Some of these queries are rather straightforward, such as "preparing a list of all hourly-paid workers who have less than two years of seniority." Other queries are more complex, requiring files to be merged for instance.

Report Preparation Applications. This category of applications differs from the previous one in that the applications it includes generate reports after having performed calculations on the data extracted. However, the two categories have much in common. Both types of applications were most common in all firms. A large proportion of these applications were run on a regular basis. Two sources of data exist for these applications: files that users created themselves, and files (or data bases) made available by DP. In the latter case, when users were asked why they did not ask DP to include the preparation of these regular reports in the production systems, users provided two types of answers. First, they said it would take months for DP to incorporate these few lines of code in a production system, and in most cases users would have to justify the new report.

Second, if something has to change in report formats, or computations, a formal request for change would have to be sent to DP; once again it would take much time to have the changes made. For such applications GIS, EASYTRIEVE, and SAS were most often used.

Graphics. The sixth category of applications are graphics, which are generally prepared using packages, such as GDDM or SAS/GRAPH. Only four of the firms studied (B, G, H, and I) provided graphics packages, and the introduction of those packages was new in those firms. At G for instance, FOCUS was introduced during the week the interviews for this study took place. Graphics do not represent a large proportion of the applications developed by users. However, those users who do develop this type of applications claim important time savings due to the use of graphics packages.

With respect to all categories of applications, it appears that some users become proficient enough with a tool that they will extend its capabilities and use it to develop applications which are not the primary purpose of the tool. Two examples are given to illustrate this.

ADRS-II is usually described as a report tool generator. However, some users will use it as a tool to answer "what if" types of questions. On the other hand, MODELER is a tool for developing financial models. However, some users stretch its capacities in order to prepare graphics with it.

5.1.5 Summing Up Background Information

In summary, the organizations studied are large firms, where UDA is supported by DP in varying degrees. In seven of the firms there exists a formal group whose responsibility is to provide assistance to users who develop applications. In five of these seven firms (A, B, D, F, I) the group is called Information Centre (IC) whereas at G it is called User Computing Facilities and at H Timesharing Support Group. However, in both G and H the groups have the same characteristics as ICs, and informally, are often called IC. In three firms, C, E, and J, some tools are offered to users, and support is provided by "product coordinators."

The firms which have an IC type of support group provide a larger number of tools than the firms where the support is provided by product coordinators. In most of the firms studied there exist tools which are available by DP but which are not supported. These tools are generally used by special categories of users who have special needs.

In the 10 enterprises studied, support provided to users has four basic components: (1) training (2) debugging assistance (3) technical assistance, and (4) administrative tasks. Other background support activities are performed in some organizations; such activities include assisting users in planning for their use of computer resources, promoting the services of the IC, helping users prepare a business case for acquiring a terminal, and so on. As a general rule, support staff does not develop code for users. However, in some

instances, consultants (or product coordinators) are formally responsible for doing some application development for users.

In the 10 organizations studied, more than 60 users were interviewed. Those users were found to fall into one of the following three categories: (1) Micro DP Departments, (2) Staff Analysts, and (3) Opportunity Seekers. The prior computer background of the users interviewed varied from absolutely no knowledge of computers and of programming to formal computer science training and several years of application development experience.

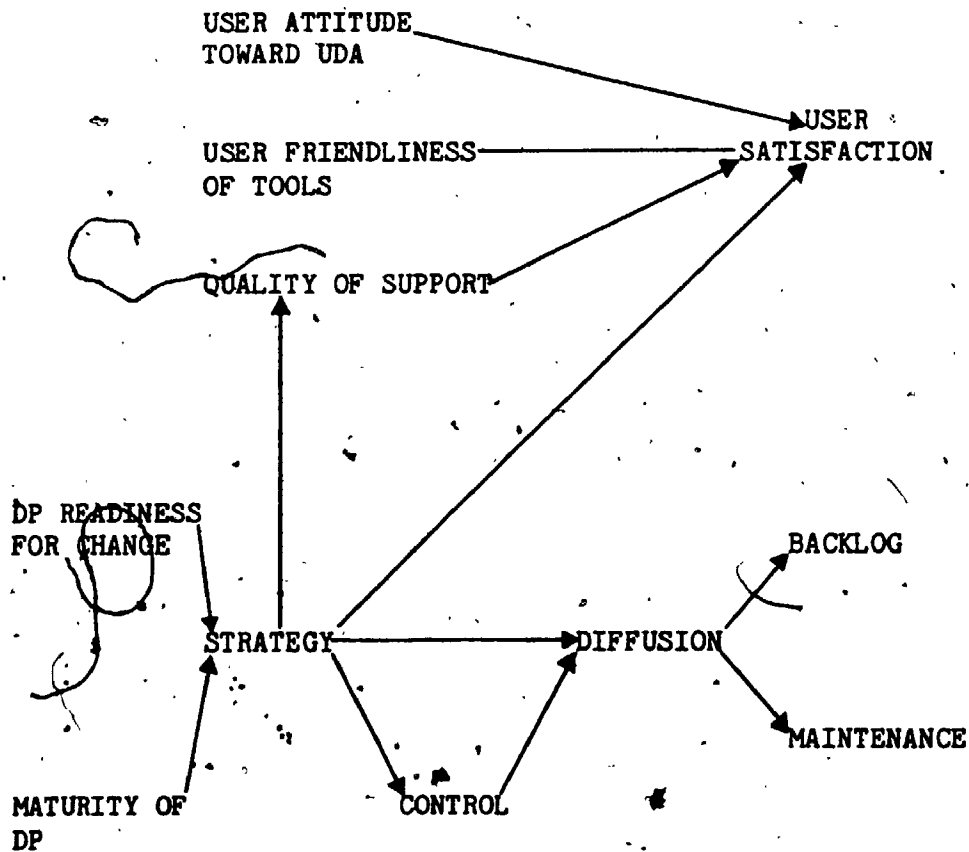
Six categories of user developed applications were identified: (1) models, the parameter of which, are changed to answer "what if" types of questions, (2) transaction processing applications, (3) data analysis applications, (4) applications which extract data, perform calculations on these data, and produce reports, (5) applications which extract data and produce reports without performing calculations on the data, and (6) applications which prepare graphics.

5.2 EVALUATION OF THE INITIAL MODEL

While the previous section described UDA in the organizations studied, this section reviews the initial research model and evaluates it with respect to its appropriateness in describing and understanding UDA in these organizations. Usefulness of the model in explaining success of UDA is discussed as well. For the reader's ease of reference, Figure 5.1 presents the model which was described in Chapter

III.

FIGURE 5.1
INITIAL RESEARCH MODEL



Each variable of the research model is analyzed below, in the following order: success of UDA, strategy, DP readiness for change, control, degree of maturity of DP, user support, user attitude toward UDA, and user friendliness of the tools.

5.2.1 Success of UDA

Recall from Chapter III, this study defines success of UDA from two perspectives: from a DP point of view and from a user point of view. The following paragraphs discuss the findings of the interview phase of the study with regard to these two aspects of UDA success.

Success from DP Point of View

Drawing on the available UDA literature, the initial research model suggests that, from the DP point of view, UDA success is measured by the degree of diffusion of UDA, which as it increases, contributes to the decrease of the DP application backlog and of the maintenance load. However, during the field work it was found that this is not the way DP managers define UDA success.

Since the definition of the variable "success from a DP point of view" changed quite dramatically due to the findings of the field investigation, some details on how the change took place are provided below.

Prior to commencing the field work phase of the present study, a questionnaire was developed, parts of which were to measure the three components of UDA success, as seen by DP (Appendix E contains the complete questionnaire which was used in this phase of the research [8]). However, when these questions were asked of the Systems Development Services Manager at A (to whom the IC Manager reports), it became apparent that the decrease in the application backlog and in

the maintenance load was not measured at A. This manager felt that UDA might have contributed to a decrease in the maintenance load. However, other "productivity improvements" such as TSO and structured programming had been implemented in DP since the introduction of the IC. Since the impact of these techniques on the maintenance load had not been measured, it would have been an impossible task to isolate the impact of UDA.

In order to pursue the matter further, the Manager of Operations and Systems Development at A was interviewed. This manager clearly stated that for DP at A, success of UDA is not defined in terms of diffusion, decrease of the backlog and of the maintenance load. When asked how he defined success of UDA from a DP point of view, the manager stated that in order to say that UDA is successful, DP has to be able to demonstrate to top management that users use the computer in a way which is profitable to the company. This manager also identified user satisfaction with the services provided by the IC staff as being important to DP. However, the true measure of success remains the favorable cost benefits to the user community.

Even though the definition of UDA success provided by A was very different from the research model definition, the latter was not abandoned yet. In the second company studied, B, the Director of DP was first asked how he defined success of UDA. The manager indicated that UDA success was measured in terms of increase in user productivity and in improvement of decision making due to the applications users develop themselves. In response to the same question, the Office

Systems manager added that one of his responsibilities was to provide top management with figures demonstrating the extent to which UDA is profitable to B. For both managers, user satisfaction with the services provided by the IC was also mentioned as being important to DP. Yet as in the case of A, the cost benefits aspect of UDA is the critical dimension of success.

At company D (the next company studied after B), user satisfaction was identified by the DP manager as being an indication of UDA success. The ability to demonstrate to top management that UDA is profitable to the organization was identified, by the IC manager, as being critical in order to justify the necessary increases in resources devoted to UDA support. As was the case in the other two firms, while it was felt that UDA might have had an impact on the maintenance load, the impact was not measured, and more importantly, it was not identified as a dimension of success of UDA.

The same pattern of responses was obtained in the remaining seven organizations studied: in none of them were degree of diffusion, the decrease in the backlog and in the maintenance load, identified as being measures of success from a DP point of view. Table 5.6 lists the definition of UDA success for the 10 organizations in the sample.

As indicated in Table 5.6 user satisfaction is seen by some DP managers as the primary indicator of UDA success. In most cases user satisfaction interpreted to mean "no complaints," and satisfaction with the services provided. For the Vice-president, Systems Development at

TABLE 5.6

DP DEFINITION OF UDA SUCCESS

<u>COMPANY</u>	<u>PRIMARY DEFINITION</u>	<u>SECONDARY DEFINITION</u>
A	Assurance that users use computer resources in a manner which is profitable to the firm.	User satisfaction with IC services
B	Improvement in user productivity and in decision making outcomes due to UDA.	User satisfaction with IC services.
C	User satisfaction	
D	User satisfaction	Assurance that users use computer resources in a manner which is profitable to the firm.
E	Improve user productivity	
F	Improve user productivity	
G	User satisfaction	Decrease outside timesharing usage. Assurance that users use computer resources in a manner which is profitable to the firm.
H	Decrease outside timesharing usage.	User satisfaction Assurance that users use the computer in a manner which is profitable to the firm.
I	Assurance that users use the computer resources in a manner which is profitable to the firm.	Decrease outside timesharing usage.
J	UDA should be profitable for the firm (Low cost, high user benefits)	Reduction of the number of small one-shot type of requests.

G, user satisfaction means that other Vice-presidents or senior managers no longer call to ask if they can "borrow" a programmer for a special project. To this manager, the fact that such calls no longer take place is a primary indicator of success.

For one of the firms studied, H, the primary measure of success is the decrease in outside timesharing. H implemented a Timesharing Support group in order to repatriate outside timesharing usage. This was also a motivator for G and I, and decrease in outside timesharing usage is mentioned by DP management in these two organizations as a secondary measure of success.

Table 5.6 strongly suggests that DP managers do not perceive UDA as bringing benefits directly to DP. For seven organizations out of 10 (A, B, E, F, H, I, J) the primary measure of success is the dollar benefits to users (or the firm as a whole in the case of H) brought by UDA. For the remaining four organizations, user satisfaction is the primary measure of success. It might be argued that user satisfaction indirectly benefits DP. This is true especially in the situations where DP supports UDA with a formal group (IC or others); in such cases, user satisfaction with DP services in the context of UDA enhances the image of DP in the organization as a whole. The same reasoning can be made with regard to favorable cost benefits of UDA. By creating an IC or a formal support group, DP devotes some of its own resources to UDA (personnel) as well as corporate resources (computer

capacity). As put by the Vice-president Systems Development at G, it would be most embarrassing for DP to come to the steering committee with a demand for increasing the computer capacity by \$6 million to support UDA, if DP had no evidence that the UDA activities are profitable for the firm [9].

The DP managers interviewed were asked how important an impact UDA might have on the maintenance load and on the backlog. With respect to the maintenance load, in many instances (A, B, D, F; H) it was felt that UDA had had (or could have) some effect on the perfective maintenance load. However, in these firms, maintenance was not actually measured and more importantly, the decrease of the maintenance load was not perceived by the DP departments as a measure of UDA success. In none of the seven organizations where the application backlog is large (A, B, D, F, G, H, I) was it believed that UDA would impact DP backlog. That is, the latent demand for new application is so high that reduction of the backlog is effectively impossible. For the DP Manager at B, only new technologies and techniques implemented within Systems Development have the potential of reducing the backlog. For the DP Manager at D, not only is UDA not likely to reduce the backlog, but chances are that it will serve to increase it. This increase will occur because users, having tools to query corporate files, will ask for more "mainframe systems necessary to provide them [users] with the data they need." At C, where the backlog is currently low and where DP is considering the implementation of an IC, the Systems Planning & Development Manager was asked if he thought that the IC would contribute to further reducing the backlog. His response was

much similar to the one given by the DP Manager at D, that is, that the backlog is more likely to increase rather than decrease. In the other two firms where the backlog is low (E and J) UDA is not seen as being responsible for the low level of the backlog. At E, it is felt that DP has been able to keep pace with user demand, while at J the DP Manager identifies the current unfavorable economic situation as being the main reason why users do not ask for new systems [10].

FIGURE 5.2
SUCCESS OF UDA, AS SEEN BY DP

	Favorable Costs, Benefits of UDA	User Satisfaction
PRIMARY DEFINITION	7	3
SECONDARY DEFINITION	2	3
TOTAL	9	6

Figure 5.2 summarizes the discussion of UDA success from a DP point of view in the firms studied. The Figure was derived from Table 5.6 and should be read as follows. Favorable cost benefits to user community is used as a general term which includes: improved user productivity, assurance that users use the computer resources in a manner profitable to the firm, and decreased outside timesharing usage.

Seven of the 10 organizations identified monetary benefits to the user community as the primary measure of success. The other three organizations identified user satisfaction as their primary measure of success. Of these three, two put monetary benefits as a secondary measure. Of the seven organizations for which monetary benefits is the primary measure of success, three identify user satisfaction as a secondary measure. In total, 9 of the 10 DP departments see favorable cost benefits to the user community as a dimension of success while six out of 10 mention user satisfaction.

In the remainder of the present study, success of UDA from the point of view of DP will then be assumed to be favorable cost benefits to the user community together with user satisfaction.

Success from a User Point of View

The 62 interviews conducted with users who develop applications in the 10 organizations studied confirm that user satisfaction is an appropriate measure of UDA success, from a user point of view.

In order to further explore the concept of user satisfaction with UDA, as well as the variables which contribute to explain it, users were asked to express their opinion on, and to identify the benefits of, UDA.

While, as discussed in the previous section, favorable cost benefits of UDA was most important for the DP managers interviewed, users did not identify this aspect of UDA as a primary concern. This

is not to say that users were not concerned at all with it. When asked if productivity improvements resulted from the applications they had developed, most users responded that the applications either saved (themselves or other users) several hours of work or performed tasks which were not feasible otherwise. During the interviews, it seemed that users assumed that such productivity improvements (or better decision making type of benefits) were resulting from the applications they developed. To the users interviewed, the concept of application development by end users was more important than the applications per se.

The results of user interviews contributed to identify new variables which help explaining user overall satisfaction with UDA. These variables are (1) user satisfaction with independence from DP and (2) user satisfaction with UDA environment setup. Those variables are discussed below.

User Satisfaction with Independence from DP. During user interviews, it became clear that for some users, a most important benefit of UDA was to "free" them from the problems which may be encountered when applications are developed by DP. User comments which were grouped under the heading "user satisfaction with independence from DP" pertain to four aspects of UDA. Those aspects are (1) the timeliness with which applications are developed, (2) the absence of communication problems between user and DP professional, (3) the increase in user control over the application development process, and (4) the reduction of development costs. Below are some user statements

which illustrate the meaning of the phrase user satisfaction with independence from DP.

"What I like about having APL available on the mainframe is that I am never told what to do by DP people who know nothing about my job."

"I value independence very highly. DP is very bureaucratic. With my terminal in my office, I can use the computer when I want, for what I want."

"When you develop applications yourself, you are the one who decides what is important. With DP, what is important for you is not important for them."

"The main benefit of developing applications myself? Timeliness. You get what you want when you want it."

"DP people do not understand the background of the system they develop."

"When you develop an application yourself, you control the results."

"I wrote down the time for developing this particular application: 40 hours. DP would have taken eight hours just to cost it out, and would have charged me \$42.50 an hour to do so."

"For our department it is much cheaper to have me write the EASYTRIEVE programs. DP charges \$50 an hour."

"It is much less costly to develop these models myself than having somebody from DP develop them."

"One day I asked DP to change the title in a report. It took them 7 hours and they charged me \$350; I'd rather do it myself."

"It cost my department \$400 to change a division's name and address on a report. Now we can do these changes ourselves and it surely does not cost that much."

The fourth aspect of user satisfaction with independence from DP, that is, reduction of development costs is relevant to contexts where users are charged for systems development. This was the case in firms,

A, B, C, D, E, G, and J. The average charge for systems development in these organizations is \$50 per hour. In each of the seven firms, there were users who claimed that UDA was saving money for their department; since very few of those users who develop applications are paid \$50 per hour.

However, some less enthusiastic users have suggested that the cost of the applications they develop might not be lower for the organization as a whole. Because of their lack of expertise users may take much longer than DP professionals to develop an application, and the program itself may be inefficient. An example of such a situation is the case of an inexperienced and "computer-intimidated" user who worked for two months on a GIS application. According to this user, it would have taken less than a week for a DP professional to develop the same application. However this type of application would have been given a low priority by DP, and the user department would have had to wait several months to have it developed.

User Satisfaction with UDA Environment Setup. The phrase user satisfaction with UDA environment setup includes two aspects of UDA which were most often mentioned during user interviews. Those aspects are the availability of terminals and the computer response time. Some of the statements users made to illustrate the importance of the environment setup are listed below.

"There are no terminals available near my department. I have to go all the way to the Information Centre and sometimes all the terminals there are used. One day I will have a terminal

of my own."

"They asked me if I wanted a terminal in my office. I said no. I don't want all the other people in the department coming to my office to work on the terminal while I am busy doing something else."

"I spend between 30 to 40 percent of my time on application development. We don't have a terminal at our own plant. I have to come here [the IC]. It is very time consuming. We should have a terminal."

"What is most important when you develop you own applications is the proximity of terminals."

"The main advantage [of UDA]? Timeliness. The main disadvantage? The time you have to wait to get a response back on the terminal."

"I don't even ~~try~~ work on the terminal between 10:00 and 3:00. The response time is really bad."

5.2.2 Strategy

In the initial research model, the use of the word strategy refers to the mode of inception of UDA in an organization. As discussed in Chapter III, the initial model identifies two types of strategies. A Push strategy exists when DP initiates the concept of UDA and a Pull strategy exists when users start developing their own applications without being encouraged to do so by DP. Mixed strategies were defined as being situations where there is a Pull from the user community as well as a Push from DP.

The strategy concept, as defined in the initial research model, did not prove very helpful in understanding UDA as it exists now in the firms studied.

The following example illustrates why the strategy concept has not proven to be very illuminating. At H, the Timesharing Support Group was created in order to slow the growth of outside timesharing usage. According to the above definition the strategy at H is a Pull since UDA was introduced by users first. However, nowadays at H DP makes programming tools, computer resources, and assistance available to users. For that portion of the user population which has started developing applications since the implementation of the Timesharing Support Group, the strategy is a Push. The term mixed strategy, as defined above, might be appropriate to describe UDA at H. However, at H as well as in the other 9 organizations studied, it was found that what is important is not really who initiated the concept of UDA, but more importantly the equilibrium between what users want and what DP offers. This concept of equilibrium (or goodness of fit) is illustrated below.

The DP department at D has implemented an IC and made available, through the IC, the tools GIS, ADRS-II, and PLANCODE. FORTRAN is also available but not through the IC. As in most other firms, the IC consultants at D are responsible for maintaining the technical environment of UDA, which includes deleting "keep files" [1]. The Systems Development group is responsible for programming and maintaining the production systems (which are programmed in COBOL). IC users are not allowed to alter production files. As a corporate policy, users are allowed to use the corporate timesharing services if DP in their own division cannot fulfill their needs. If neither the corporate timesharing services nor their division's DP department can

provide them with what they require, users can go to outside timesharing firms or buy minicomputers. Most users have terminals available in their working area.

At J, DP makes some of the features of ASI-ST available to users (the file creation and file update features are not made available). One coordinator is in charge of supporting ASI-ST. For approximately 18 months, IFPS has been available to users via the parent company's corporate timesharing services. In the very near future it will be available in-house. One coordinator is in charge of IFPS support. ASI-ST users are responsible for having their cards punched and submitted to DP.

Some users at D argue that DP does not give enough freedom to users to develop the applications they want. They also state that DP puts too much emphasis on security, does not provide enough tools, and so on. One individual suggested that DP should let the users alter themselves the (COBOL) code of production programs. On the other hand, at J, of the users interviewed none have mentioned that they felt that DP did not provide enough tools, put too much emphasis on security, and so on.

Given the above reporting on users' interviews only, one would suggest that D lies more toward the Pull end of a Push-Pull continuum than does J. However, given only the description of how UDA takes place in each firm, one would tend to suggest that D is toward the Push end of the continuum, as compared to J. Finally, if both the

description of UDA in each firm and the reporting of interviews are taken together, it would appear that UDA at J is in equilibrium, while it is not so at D.

What can contribute to the explanation of the disequilibrium at D? As explained by a user, there is a lack of fit between what he and some other users demand and what DP offers. This user also stated that for another portion of the population (less demanding users) the fit seemed to be good. The same phenomenon was observed in most of the 10 organizations studied.

These observations argue for the inclusion of the concept of goodness of fit between what users want (amount of Pull) and what DP offers (amount of Push) in a revised model of UDA success.

5.2.3 DP Readiness for Change

The initial research model stated that "a Push strategy is more likely to take place in situations where DP has felt a need for change." As discussed in the previous section, the interview results suggest that the concept of goodness of fit between what users want and ask for (amount of Pull) and what DP offers (amount of Push) is more appropriate here than the concept of strategy. Nevertheless, the variable DP readiness for change remains relevant in that the results of the interview phase suggest that DP readiness for change has an impact on the amount of Push.

The initial research model defines DP readiness for change as follows. The senior DP manager perceives strong pressures, both from the user community (external environment) and from DP itself (internal environment). Moreover, the gap between the demands of the environment and the current performance of DP has to be perceived as having significant adverse consequences for DP itself. During the field investigation it was found that in addition to (or in conjunction with) the above, the DP manager's attitude toward the concept of UDA was a useful indicator of DP readiness for change.

Once the interview phase was completed in each of the firms studied, DP was classified in one of the following two categories: (1) higher degree of DP readiness and (2) lower degree of readiness. The results of this classification are as follows. Firms B, C, F, G, and H, were classified in the higher degree of DP readiness category and firms A, D, E, I, and J, were classified in the lower degree of readiness category.

The above classification is based on comparison between firms. That is, it provides a relative degree of DP readiness rather than an absolute one.

5.2.4 Control

The initial research model, by including the variable control, emphasizes the importance of the intent of controls rather than of the control mechanisms themselves. It was suggested that a DP department

which does not want to encourage users to develop applications is likely to implement control mechanisms in order to limit the diffusion of UDA. Controls such as no access to the company's files and/or computer(s), no right to purchase computer time from outside timesharing firms, etc., are examples of control mechanisms which were termed UDA-restrictive. On the other hand, a DP department may want to encourage users to develop their own applications, while ensuring data security and "wise" use of computer resources. In such a situation DP might implement control mechanisms such as chargeout, copy management, security packages, and so on. According to the initial research model, the intent of such control mechanisms is non-UDA-restrictive (i.e. non-restrictive with regard to UDA).

In the initial model the variable control is important because of its hypothesized effect on diffusion of UDA, and because of its indirect effect on the decrease of the backlog and of the maintenance load. Given the change which occurred in the definition of success of UDA from a DP point of view, it appears that intent of control is no longer a critical variable.

However, the results of the field investigation suggest that the concept of control is indeed part of the UDA phenomenon. First, the interviews conducted with DP managers and IC managers suggest that many of the concerns of DP with respect to UDA belong to the domain of control. Second, the same interviews also suggest that some control mechanisms, such as chargeouts, may help DP in successfully managing UDA.

TABLE 5.7

DP MANAGEMENT CONCERNS WITH REGARD TO UDA

"Some users have started developing production systems while they should not."

"My main concern with UDA is one of control: How do you get the right tools with the right people using them in the right way?"

"My main concern is the capacity problems we may encounter if the growth [of UDA] is not controlled."

"Our main concern here is to insure security of corporate data."

"Some users develop core applications. They should not."

"My main concern is how can you make sure that the information which results from the applications developed by users is correct." (Mentioned in 5 organizations)

The DP managers as well as the IC managers of the 10 organizations studied were asked what were the main issues of UDA and what were their main concerns with respect to UDA. Table 5.7 typifies the concerns stated by the managers interviewed. Many of these concerns have a control dimension.

The concept of control is also relevant to UDA primarily with regard to the control mechanisms which may help DP in being successful, with respect to the cost benefits aspects of UDA. The case of firm B is used here to illustrate what control means in the present context. B has implemented an IC to contribute to the company's effort to improve

people productivity. The Office Systems Manager (who is in charge of the IC) keeps track of the benefits, in dollar terms, of UDA to the user community. This is done by meeting with the user management on a regular basis to obtain "hard numbers" on the number of hours saved, increased profit due to better decision making, and so on. When top management enquires about the impact of the IC, the information is available to them.

According to the information gathered during the field work, it appears that it is this "keeping track" aspect of control which is the most important for DP; with regard to success of UDA. The revised version of the model, which will be discussed in Section 5.3, discusses further the relevance of this aspect of control.

5.2.5 Degree of Maturity of DP

The variable maturity of DP did not prove as useful as expected in helping understand UDA in the organizations studied. The initial research model stated that a Push strategy is more likely to take place in firms where DP is more mature, and conversely, that Pull strategies are more likely to occur in organizations where the DP department is less mature.

The objective in measuring maturity of DP was to differentiate between more mature and less mature DP organizations, rather than to determine precisely at which stage of growth a certain DP department is. The instrument used in this study was McCririck's (1979)

questionnaire, which is based on Nolan's (1979) stages of growth. Both the results of this questionnaire and the knowledge of each firm acquired during the field investigation, suggest that the DP departments of all 10 organizations studied were at a relatively high degree of maturity.

The DP profiles of the firms studied were compared to the DP profiles of "more mature and less mature" DP organizations which participated to Benbasat, Dexter, and Mantha's study (1980). Each one of the 10 DP departments studied here was found to belong to the more mature category.

In a situation such as the present one, what is required is an instrument which adequately captures the fine differences of degree of maturity between more mature DP departments. However, to date, such an instrument is not available. In a discussion of the results they obtained using McCririck's instrument, Goldstein and McCririck (1981) emphasize the "low intercorrelations obtained among the maturity variables of the Stage Hypothesis." As an explanation for such results, they argue that while "it may be that our results simply reflect the failures of our set of independent variables to capture the essence of Nolan's maturity concept...Before further empirical work is conducted, the underlying principles of the Stage Hypothesis should be critically reviewed." (Goldstein and McCririck, 1981, p.322)

Because it was not possible to capture the differences of degree of maturity between the DP departments of the firms studied, the

variable maturity of DP would have taken the same value for all 10 organizations. Since a variable taking a constant value would have been useless in understanding and explaining UDA success, the variable maturity of DP was abandoned.

5.2.6 User Support

The initial research model suggests that "the higher the quality of support users receive from DP, the more they will be satisfied with the organizational context in which UDA takes place." The interviews conducted with users confirm the importance of the variable "user support" in explaining user satisfaction. However, it was found that users had different perceptions of the quality of support. For instance, while, in a given company, some users might claim that the Information Centre staff were people oriented, other users might say that they are not. Since no objective measure exists of the quality of support provided to users, and since it is out of the scope of the present study to develop such a measure, the revised version of the model uses "user satisfaction with support" as a variable explaining user overall satisfaction with UDA.

Below are some user statements which illustrate users' perceptions of user support.

"The support you receive is the most important thing to keep you going."

"Information Centre personnel must be 'people-oriented.' "

"Whenever you have a problem you call the Information Centre and they are always ready to help you."

"I asked for help from [the product consultant] once and he didn't know how to solve my problem."

"I never ask the IC guys to help me, anyways, they know nothing about APL."

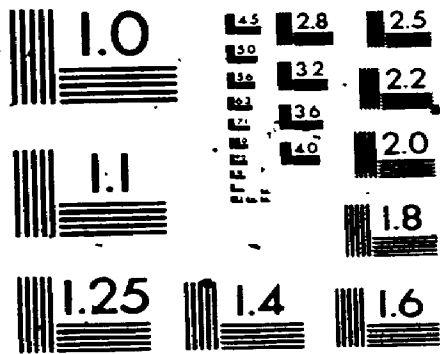
"For us, technical support is very important. Yet, nobody [from the support group] sits with you and explains how to go from TSO to CMS, and so on. When there are technical problems they [the consultants] are not always very nice."

5.2.7 User Attitude Toward UDA

The initial research model suggests that user attitude toward UDA is related to user satisfaction. The user interviews support the inclusion of this variable in the model. However, since "attitude toward UDA" is a broad concept, an effort was made during the interview phase of the study, to identify which aspect of UDA was most critical with respect to user attitude.

The information gathered during the interviews suggests that user perceptions of their own ability to develop applications and of their "self-sufficiency" in the context of UDA, because of their impact on user satisfaction, be included in the model under the heading of user attitude toward UDA. Furthermore, among the users interviewed, some appeared much confident in their ability to perform UDA activities and did not have to rely much on the assistance of DP consultants. On the other hand, other users expressed some concerns with respect to their ability to develop applications. The latter type of users often had a weaker computer background than the former. As will be discussed in a

3



following section, the computer background of users will be included in the revised version of the research model for its impact on other independent variables.

5.2.8 User Friendliness of the Tools

The relevance of user friendliness of the tools as a variable of the model was confirmed during the field investigation. In some cases, emphasis was put on the lack of user friendliness of the tools available to users. Most of these tools require relatively long learning times and much practice before a user can become proficient in their use. For instance, at D, GIS training consists in four days seminars, while at H, PLANCODE and EASYTRIEVE seminars last one week. According to several users and DP people, if a user does not practice right away with the tool, the time he spent being trained will have been wasted for him (and for the consultant as well). Moreover, as put by some users, even after having practiced quite a lot, "if you go away from it [application development] for some time, you almost have to learn the tool all over again." However, the interviews suggest that the perception of the user friendliness of a given tool is likely to vary according to the computer background of a user. As discussed in the following section, computer background of the user will be included in the research model. Below is an illustration of user statements that confirmed the appropriateness of user friendliness.

"When you don't have much knowledge about computers, learning how to use GIS takes quite a while and you have to practice."

"Using EASYTRIEVE is a pain in the neck: it is hard to learn."

"With ADRS-II, you can handle quite a large volume of data, the code is easy to modify, the report format is easy to change."

"EASYTRIEVE is very good: easy to learn, easy to use, and you can do quite fancy things with it."

"EDGE is easy to use but not flexible enough, while FORESIGHT is more flexible but more difficult to use."

5.2.9 Influence of Computer Background of Users

While the above paragraphs discuss the relevance of the variables of the initial model, the inclusion of a new independent variable, namely, computer background of users, is discussed here.

During the interviews, it was found that computer background of users had an impact on two other variables, that is, perception of user friendliness of the tools and user attitude toward UDA. Difference in computer background of users partly explained why a given tool was perceived as being easy to use by some users and difficult to use by others. Computer background of users appears to have a similar effect on people attitude toward UDA. Users with a stronger computer background tend to have a more positive attitude toward the concept of UDA than users who have a weaker computer background.

The revised version of the research model then includes computer background of users as an independent variable.

5.3 A REVISED VERSION OF THE MODEL OF SUCCESS OF UDA

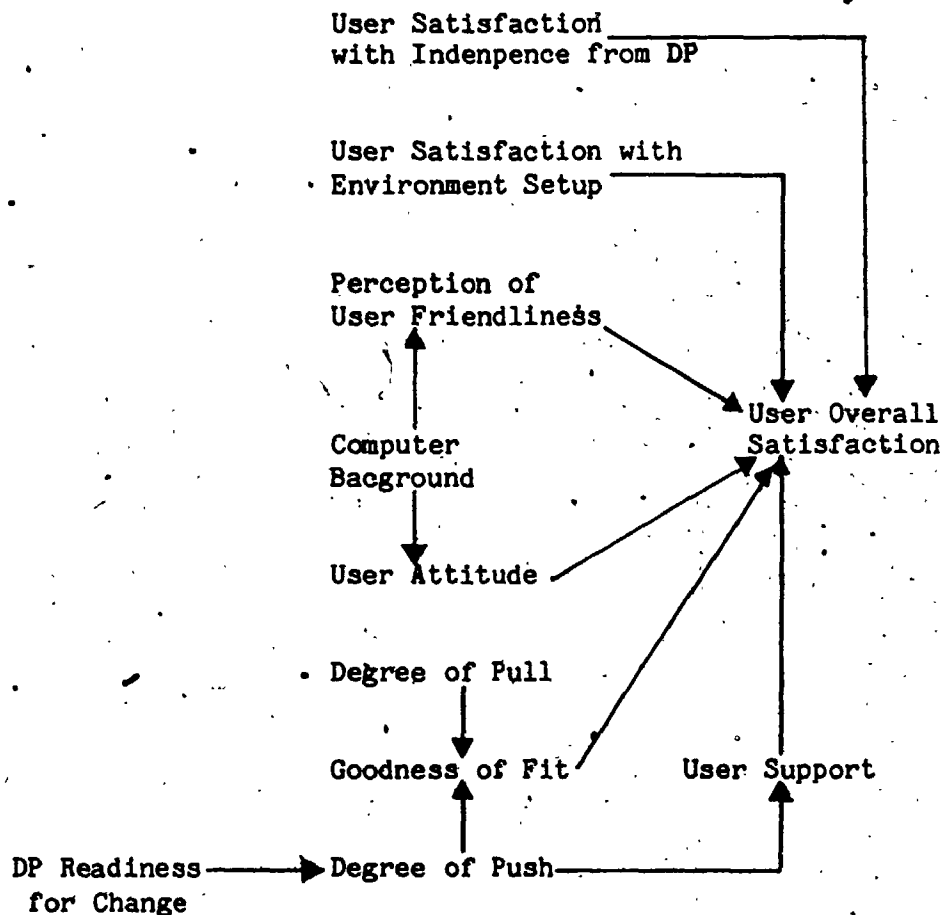
This section presents the revised version of the model of the factors of success of UDA. This version is the outcome of the interview phase of the study. The purpose of Phase III, which is to be discussed in Chapter VI, is to test parts of the model. The version of the model presented below actually consists two sub-models which, for the purposes of the present study, are treated independently. One sub-model deals with success defined as user satisfaction and is presented in Section 5.3.1. Section 5.3.2 discusses the other sub-model which defines success as favorable cost benefits of UDA.

5.3.1 "User Satisfaction" Sub-Model

The User Satisfaction sub-model is based on the results of the interview phase, and is a refinement of the initial research model. As discussed in Section 5.2, while most of the variables of the initial research model were found to be helpful in explaining user satisfaction, some were abandoned and new ones were added. This section briefly describes the relationships between the variables of the User Satisfaction sub-model and suggests a series of propositions which will be tested in the next phase of the study, that is, the survey phase. Figure 5.3 illustrates the model.

The model defines UDA success as being overall user satisfaction with UDA, and includes 10 independent variables. As implied by the model, the independent variables are not all directly related to UDA.

FIGURE 5.3
USER SATISFACTION SUB-MODEL



For instance, the model suggests that computer background of users has a moderating effect on both user attitude toward UDA and user perception of user friendliness of the tools. In turn, the model proposes that those two variables are directly related to satisfaction with UDA. Other variables, such as satisfaction with independence from DP, satisfaction with support, are directly related to overall user satisfaction with UDA. Finally, as discussed in detail in Chapter VI,

the variable goodness of fit is a function of two other variables, namely, the degree of Pull exerted by users and the degree of Push exerted by DP.

The following propositions formally describe the relationships between the variables of the model. A major purpose of the survey phase of the study, the results of which are presented in Chapter VI, is to test the propositions listed below.

Proposition 1: The higher the degree of user satisfaction with independence from DP, the higher the degree of overall user satisfaction with UDA.

Proposition 2: The higher the degree of user satisfaction with environment setup, the higher the degree of overall user satisfaction with UDA.

Proposition 3: The more user friendly a programming tool is perceived by the user, the higher the degree of user overall satisfaction with UDA.

Proposition 4: The more positive the user attitude toward UDA, the higher the degree of user satisfaction.

Proposition 5: The computer background of the user has a moderating effect on the user perception of user friendliness of programming tools.

Proposition 6: The computer background of user has a moderating effect on user attitude toward UDA.

Proposition 7: The higher the degree of user satisfaction with support the higher the degree of user overall satisfaction with UDA.

Proposition 8: The better the fit between the Pull exerted by users and the Push exerted by DP, the higher the degree of overall satisfaction with UDA.

Proposition 9: The degree of Push exerted by DP will be higher in the cases of firms with a higher degree of DP readiness for change.

Proposition 10: The perceived degree of Push is positively correlated with the user satisfaction with support.

5.3.2 "Favorable Cost Benefits of UDA" Sub-Model

All 10 of the firm studied made available UDA facilities, such as software tools, computing resources, and assistance, to user departments. In seven of the ten companies, "favorable cost benefits of UDA" was identified by DP management as being the primary definition of UDA success, while it was identified as being a secondary definition of success by DP management in two of the remaining three firms. For the firms studied, profitable cost benefits of UDA is either (1) profitable cost benefits of the applications developed, or (2) decrease in outside timesharing usage.

However, in most cases, the above definition of UDA success was accompanied by some statement pertaining to the ability of DP to demonstrate to top management that those benefits indeed exist. The following comments illustrate this point.

"Unless we can demonstrate that the applications users develop are profitable to the company, we won't be able to assert that UDA is successful. And for the time being, we cannot perform such a demonstration."

"The president comes from Finance. What he wants to see as a result of our efforts [to facilitate UDA] is a good ROI."

"As of today the Information Centre does not have sufficient [manpower] resources so as to provide a good service to users. However, we won't be able to get more until top management is convinced that UDA is profitable."

"How do I define success? User satisfaction....However, we have to have 'hard' numbers in order to assure top management that UDA is cost effective....It would be most embarrassing for us to go to the steering committee with a demand for a \$6 million increase [in computer capacity] to accommodate UDA if we cannot demonstrate that what users do is profitable for the company."

As suggested by some of the above quotes, DP is not always successful in demonstrating to top management that UDA is cost effective. From the information gathered in the 10 firms studied, the issue of UDA evaluation appears to be critical. The experience, both of DP departments which are successful in demonstrating that UDA is cost effective, and of DP departments which are less successful in doing so, suggests that the following views of evaluation are most relevant to UDA.

Evaluation is a set of planned, information gathering, and analysis activities undertaken to provide those responsible for the management of the change with a satisfactory assessment of the effects and/or progress of the change effort....[One] key word in this definition is planned. Evaluation is often glossed over and viewed as an add-on or extracurricular activity of a change effort. If evaluation is viewed as such and generally undertaken as an afterthought, it is predictable that the quality of the information or the cost of the evaluation, or both, are likely to be unsatisfactory. (Beckhard and Harris, 1977, p.86)

Evaluation is part of the wider process of implementation and begins before the system is designed. (Keen, 1975, p.19)

In his discussion of the implementation of Decision Support Systems, Keen emphasizes the importance of a negotiated contract between the consultant (systems designer) and client (user, top

management). As important is the fact that the negotiation "occurs well before the system is even designed" (p.73). Keen suggests the following as an "ideal agenda for negotiation":

1. To define "success;"
2. To allocate resources and responsibilities;
3. To develop methods and criteria for evaluation, including a consensus as to what "key indicator" may be used to test the status or accomplishment of each aim of the system. (Keen, 1975, p.23)

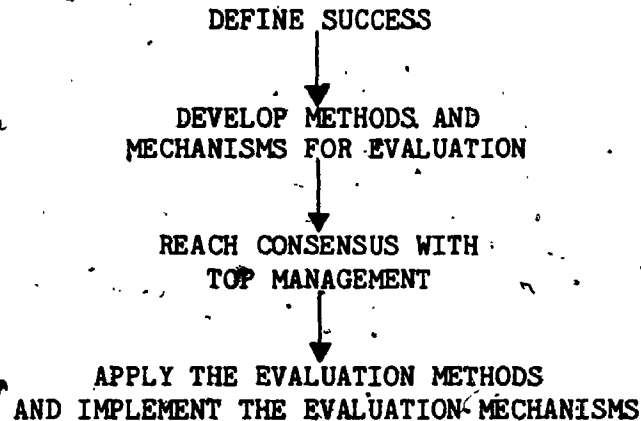
The experience of the firms studied suggests that for DP to successfully demonstrate to top management that UDA is profitable to the company, the evaluation process must begin before UDA is introduced in the organization. Moreover, it is critical that DP and top management negotiate a "contract," and that they both respect the terms of this contract.

The model discussed below borrows heavily from Keen's "ideal agenda for negotiation." Nevertheless, the interviews conducted during the study suggest that some adaptations be made. The model is illustrated in Figure 5.4 and is discussed below.

The model states that the successful management of the UDA evaluation process includes four critical steps. While those steps are presented here in a linear fashion, it is likely that overlaps exist and that iterations will take place. The four steps are (1) define UDA success, (2) develop methods and mechanisms for evaluation, (3) reach a consensus with top management with regard to (1) and (2), and (4) apply

FIGURE 5.4

"FAVORABLE COST BENEFITS OF UDA" SUB-MODEL



the evaluation methods and implement the evaluation mechanisms. In the discussion which follows, the cases of B and H are used to illustrate situations where DP successfully manages the UDA evaluation process.

Define UDA Success

For DP, the definition of UDA success is closely related to the types of pressures exerted on DP so as to move toward UDA. In the cases of B and H, these pressures were clearly identified, and UDA success readily defined. At H, the costs of outside timesharing had been doubling every year prior to the introduction of in-house Timesharing Services. Both top management and DP were concerned with the increasing costs. DP decided to introduce in-house Timesharing Services, and defined UDA success as the "slow-down of the growth of outside timesharing and eventually 'repatriate' all outside timesharing usage."

At B, DP had received the mandate, from top management, to participate to the corporate effort toward productivity increase. Providing users with UDA tools and support was identified by DP as an appropriate way of increasing user productivity. DP defined UDA success as "the increase in user productivity and in decision making outcomes."

Develop Methods and Mechanisms for Evaluation

For DP at H, the method of evaluation of UDA success is rather straightforward. The outside timesharing expenses of user departments are monitored, in particular the expenses of two or three "heavy" outside timesharing users. DP puts much effort in repatriating those heavy users.

At B, DP developed a document entitled "Information Centre Guidelines for User Business Cases." Those guidelines, based on pilot studies conducted in the firm, as well as on other firms' experience, serve to calculate the value of productivity improvements and the value of improved decision-making capabilities. Moreover, users are charged for the services they receive from the Information Centre (IC), as well as for their use of computer resources. Finally, the manager in charge of the IC meets with user management on a regular basis in order to assess with "hard numbers" the cost effectiveness of UDA.

Reach a Consensus with Top Management

For both B and H, a consensus was reached with top management on the definition of UDA success and on the methods and mechanisms of evaluation. However, to reach such a consensus requires strong arguments on the part of DP. At B for instance, DP conducted pilot studies in order to demonstrate some of the productivity improvements which could be obtained from UDA. At H, DP made several presentations to the steering committee in order to make clear how the in-house Timesharing Services would "attack" the problem of outside timesharing use.

Apply the Evaluation Methods and Implement the Evaluation Mechanisms

This step of the evaluation process comes after the UDA tools and support have been introduced. The responsibility of DP at this stage is to ensure that the evaluation mechanisms are appropriate and that they work well. In the case of B, for instance, the task of assessing the cost effectiveness of UDA requires that the manager in charge of the IC obtains the collaboration of users. While it does not, in fact, appear to be a problem at B, it was found to be more difficult in other firms (at D for instance).

While the above discussion presented the example of two firms where the UDA evaluation process was successfully managed, the following describes some difficulties two other firms have encountered in managing this process.

At company A, DP management defines UDA success as the assurance that the applications users develop are profitable for the firm. However, DP does not dispose of the appropriate evaluations methods and mechanisms. First, users at A are not charged, either for services received or resources used in the context of UDA. When the IC was introduced at A, it was decided that users should not be charged, so as to encourage them to use the IC services and tools. Since they do not have to pay for the services they use, users are not motivated to determine the cost effectiveness of the applications they develop. Second, no formalized mechanism exists where users are formally invited to assess the cost effectiveness of their applications.

The latter is a difficulty that D also has to face. While users at D are charged for their use of the computer resources, there is no formal mechanism by which they have to assess the "profitability" of the applications they develop. In this case, DP and the IC bear the "fardeau de la preuve" but do not have the authority for making users assess the value of their applications.

The cases of C and J are different from all the cases discussed above. At C and J, users are charged, in real dollars, for all their use of DP resources. In both firms users are responsible for demonstrating to top management that UDA is profitable to the firm. In those situations, the problem of UDA evaluation is not as acute as it is in firms where DP bears the responsibility of such a demonstration.

The model discussed in this section helps better understand the process of UDA evaluation in the organizations studied, as well as to explain why some DP departments are less successful in managing the evaluation process.

5.4 SUMMING UP THE INTERVIEW PHASE

The interview phase of the study had two main objectives: (1) to better understand UDA and the various organizational contexts in which it can take place, and (2) to evaluate the initial research model with respect to its appropriateness in describing and understanding UDA.

This chapter first provided some background information on UDA in the 10 companies studied. All 10 firms made available UDA tools and support. The background information on the firms includes the organizational and physical location of UDA support, DP reasons for introducing UDA tools and support type of assistance provided to users, etc. A taxonomy of users, as well as a taxonomy of the applications users develop, were proposed.

This re-examination of the initial research model in the light of the information gathered during the interviews led to (1) redefining success from the point view of DP, (2) abandoning the variable maturity of DP, (3) adding the variables user satisfaction with independence from DP, user satisfaction with environment setup, and computer background of user, and finally (4) replacing the variable strategy (mode of inception of UDA) by the variable goodness of fit between the

Push exerted by DP and the Pull exerted by the user.

A revised version of the research model was proposed. This model actually consists of two sub-models which, for the remainder of the present study are treated separately. The first sub-model, the User Satisfaction Sub-Model, is a refinement of the initial research model. The Favorable Cost Benefits of UDA Sub-Model puts emphasis on the importance, for DP, of successfully managing the UDA evaluation process. The model suggests four critical steps in UDA evaluation. Those steps are: (1) define UDA success, (2) develop methods and mechanisms for evaluation, (3) reach consensus with top management, and (4) apply the evaluation methods and implement the evaluation mechanisms.

The final phase of the study, that is the survey phase, was aimed at testing the propositions embodied in the User Satisfaction Sub-Model. The results of this phase are presented and discussed in Chapter VI. The Favorable Cost Benefits of UDA Sub-Model will not be examined further in this study.

FOOTNOTES

[1] Before going on to Section 5.1, the reader should realize that the linear mode of presentation for the results of Phase II is not representative of the process that took place in order to arrive at these results. That is, the organization of this chapter suggests that the research model was revised once all data were gathered. What actually took place was an iterative process, where data gathered in a firm were immediately compared to data gathered in the previous organizations, and where several versions of a model of the factors of success were developed and abandoned. However, it does not seem appropriate, or even feasible, to present the results of Phase II in a format that would reflect this iterative process.

[2] As mentioned in Chapter IV, since some of the participating organizations expressed the desire that their names be held confidential, all the participating firms are labeled with a capital letter only.

[3] G has a User Computing Facilities group and H a Timesharing Support Group. While these groups do not have the IC label, their organization, purpose, and mode of operation are essentially those of an IC. For the sake of simplicity, in the remaining of this section, these two groups will be included under the generic label IC.

[4] Martin (1982, pp.20-24) provides a list of nearly 100 "products," indicating for each product the type of tasks which can be performed. Table 5.4 uses the same task categories as Martin.

[5] Scenario H, in Appendix F discusses this issue with regard to the conversion of outside timesharing usage. Moreover, in Section 5.2 of this chapter this issue is also discussed.

[6] According to the IC Supervisor, one reason for which the IC budget is "frozen" is that top management is not convinced that the use of the computer by users is actually profitable for the firm. This issue will be discussed later.

[7] MODELER (name disguised) is modeling software package developed in-house at C.

[8] The questions pertaining to diffusion, the decrease of backlog and of the maintenance load are QUESTION B1., QUESTION B3., and QUESTION B4., and QUESTIONS C4. through QUESTION C12. Lientz and Swanson's (1980) questionnaire was used as a basis for developing the questions on the decrease of the backlog and of the maintenance load.

[9] It might be argued here that the responsibility for demonstrating UDA profitability to the firm belongs to the user community. However, in most of the firms studied, DP was held responsible for it.

[10] Users at J are charged in real dollar terms for the DP services they receive. Given an unfavorable economic environment, users have to reduce their expenses as much as they can.

[11] Keep files are permanent files, that is, files that users saved. In most of the firms studied, the support staff is responsible for deleting those keep files.

CHAPTER VI

PRESENTATION AND ANALYSIS OF THE QUESTIONNAIRE DATA

This chapter focuses on Phase III of this study, namely, the survey phase. The purpose of this phase was to both obtain additional information on UDA from a large number of users, and to gather data in order to test the relationships suggested by the User Satisfaction Sub-Model discussed in Section 5.3.2 and illustrated in Figure 5.3.

Section 6.1 focuses on the research methodology, from questionnaire design to test for non-response bias. Section 6.2 presents the questionnaire results in a descriptive fashion. Section 6.3 first addresses bivariate analysis of relationships between variables of the research model, and then presents the results of regression analysis conducted in order to test the model of the factors of success.

6.1. METHODOLOGICAL ASPECTS OF PHASE III

The following paragraphs pertain to the major methodological aspects of Phase III. The research questionnaire is presented first, in conjunction with questionnaire design and testing. The distribution of the questionnaire, the response rate achieved, and the study of non-response bias are also discussed.

6.1.1 The Survey Questionnaire

The survey questionnaire was used to obtain additional descriptive data from a large number of users, as well as to gather data in order to test the research model. Since the survey instrument was going to be sent by mail and completed by respondents themselves, it was important to arrive at a questionnaire which was easy to fill out, with as unambiguous questions as possible.

A first version of the questionnaire was prepared prior to beginning the interview phase of the study. This initial version included the same five sections as the final questionnaire (the final version of the questionnaire is in Appendix E). Section I of the questionnaire provides users with definitions of terms for which ambiguities might have existed. Section II is entitled "General Questions About Application Development" and is aimed toward gathering descriptive data which are to be used to draw a profile of users. Section III focuses on users' evaluations of the context within which they develop applications, while Section IV is devoted to the evaluation of the main programming tools users use. Finally, Section V pertains to basic demographic and background information.

Information gathered during the first few user interviews of Phase II was found useful to test for the appropriateness of the questionnaire "content." That is, these interviews suggested the inclusion of new questions or items, such as satisfaction with computer response time and terminal availability, as well as satisfaction with

independence from DP. Interviews with users and with DP managers also suggested the inclusion of questions Q16 to Q18 of Section II. These questions are aimed at measuring the improvement in user productivity due to the applications users develop themselves. From these suggestions, a new version of the questionnaire was prepared.

The third stage of questionnaire design consisted of a pre-test in which six users from three different firms completed the questionnaire. When an ambiguous point arose, the user was asked to explain what was not clear, but no elaboration was given to him at that point; that is, the user had to fill out the questionnaire as if he had received it in the mail. However, once the questionnaire was completed, ambiguities were discussed, and the respondent asked to indicate what wording he thought would make the question clearer. Users were also invited to comment on all aspects of the questionnaire, such as format, length, etc. At the time of conducting this pre-test, the questionnaire had the same number of questions as it has now. Yet, it had 25 stapled sheets. One frequently voiced comment was that, even if it did not actually take that long to complete, the questionnaire was so thick that it appeared much longer.

The questionnaire was revised so as to include suggestions for improvement. It was photoreduced and printed in a booklet form. Two users were asked to fill out this newer version and did not indicate any difficulties.

6.1.2 Questionnaire Distribution and Response Rate

Questionnaire Distribution

The questionnaire was distributed to users in the 10 organizations studied. In each of the 10 firms the DP department was asked to provide a list of users who developed applications. In most cases, DP had a list of users who had an account number with the Information Centre (IC) or with the timesharing services, but did not know which users actually developed applications and which merely ran applications developed by others. In such cases, DP would provide a list of all users having an account number. In other cases, such as C, H, and J, DP provided a list of only those users who were thought to be developing applications. Table 6.1 indicates the number of questionnaires sent to users in each company, as well as the status of these users, as indicated by DP. A total of 1074 questionnaires were sent.

The questionnaire was sent along with a covering letter (see Appendix E) explaining the purpose of the research project and ensuring users that their responses would remain confidential. Respondents were also provided with a postage-paid return envelope. A Reminder (see Appendix H) was sent to users two weeks after the questionnaire had been mailed. The reminder emphasized the importance for the research project of obtaining information from UDA users.

TABLE 6.1

NUMBER OF QUESTIONNAIRES DISTRIBUTED PER FIRM

<u>COMPANY</u>	<u>NUMBER OF USERS</u>	<u>USER STATUS</u>
A	150	Users who have an account with the IC
B	128	Users who have an account with the IC
C	58	Users who are thought by DP to be developing applications
D	129	Users who have an account with the IC
E	64	FORESIGHT users
F	90	Users who have an account with the IC
G	100	Users who have an account with the Personal Computing group
H	42	Timesharing users who are thought by DP to be developing applications
I	250	Users who are thought by DP to be developing applications
J	63	Users who are thought by DP to developing applications
TOTAL:	<u>1074</u>	

Response Rate

A response rate of 40% was obtained in Phase III. As illustrated in Table 6.2, the rate varies across organizations. Companies B and H had the highest rate (59%) while firm I had the lowest rate, 22%. Such a low response rate at I might partly be explained by the fact that DP included, with the questionnaire, a covering letter of its own. This covering letter, first mentioned to respondents that DP had approved the sending of the questionnaire. However, it also pointed out to users that DP was aware that the difficult economic environment the company was facing "may make it inadvisable" for the users to take the time to complete the questionnaire. If this was the case, users were advised to "dispose of this correspondence and of the questionnaire form." This covering letter probably contributed to the lower response rate from users at I.

While the overall response rate was 40%, the "usable" response rate was 25%; that is, data from 272 questionnaires were used in the data analysis. Table 6.3 illustrates the sources of the 15% difference. Eighty-three users returned their questionnaire along with a note indicating that while they had an account number with DP they did not actually develop applications. Some indicated that they had received training with one or several tools but had not found the time to develop applications or had found that the tool did not meet their needs. Others indicated that they merely ran applications developed by others. Eighteen questionnaires were rejected because they had been filled out by DP professionals working in a DP department.

TABLE 6.2
RESPONSE RATE PER FIRM

<u>COMPANY</u>	<u>QUESTIONNAIRES SENT</u>	<u>QUESTIONNAIRES RETURNED</u>	<u>RESPONSE RATE</u>
A	150	52	35%
B	128	75	59%
C	58	36	52%
D	129	58	45%
E	64	24	38%
F	90	35	39%
G	100	32	32%
H	42	29	59%
I	250	55	22%
J	63	33	52%

OVERALL RESPONSE RATE: 40%

As mentioned above, the lists provided by most DP departments were lists of users who have an account number with the IC, or with timesharing services. Table 6.3 suggests that several users have an account number but do not actually develop applications, and that some DP professionals (from DP) also use the IC services and programming tools.

Fifteen questionnaires were found to be incomplete, and eight were returned with an indication that the addressee was unknown. Three

TABLE 6.3
RESPONSE RATES

	<u>NUMBER</u>	<u>PERCENTAGE</u>
TOTAL NUMBER OF QUESTIONNAIRES SENT:	1074	100%
TOTAL NUMBER OF QUESTIONNAIRES RETURNED:	429	40%
Non-usable questionnaires:		
Users who do not develop applications themselves:	83	
DP professional working in DP:	18	
Incomplete:	15	
Engineering applications:	3	
Returns to sender:	8	
Usable but received after cutoff date:	30	
TOTAL:	157	15%
TOTAL NUMBER OF QUESTIONNAIRES USED:	272	25%

respondents (from I) returned the questionnaire not completed, having indicated that they were developing engineering applications as opposed to administrative applications.

Finally, 30 usable questionnaires were received after the cutoff date, that is, three weeks after the reminder had been sent to users. These 30 questionnaires were used in the test for non-response bias, discussed below.

Table 6.4 outlines the distribution of the 272 questionnaires across the 10 organizations studied.

TABLE 6.4
DISTRIBUTION OF RESPONDENTS ACROSS FIRMS

<u>COMPANY</u>	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE OF THE SAMPLE</u>
A	28	10.3
B	40	14.7
C	42	15.4
D	27	9.9
E	15	5.5
F	22	8.1
G	17	6.3
H	21	7.7
I	20	7.4
J	40	14.7
	----- 272	----- 100.0%

NOTE: The term respondent pertains only to those users whose questionnaire responses were used in the data analyses. That is, while 429 people returned the questionnaire, there are 272 respondents.

6.1.3 Study of Non-Response Bias

Non-response bias is a concern inherent to all survey research. Oppenheim suggests the following: "To study response bias, we must make sure that we know the return date of every questionnaire, for it has been found that respondents who send their questionnaire very late

are roughly similar to non-respondents" (Oppenheim, 1966, p.34). In order to test for non-response bias, Oppenheim suggests comparing "early respondents with late respondents (in terms of their answers to the questionnaire)" (ibid.).

In the present study, questionnaires returned after the cutoff date (30 questionnaires) were put aside and their responses compared to the 272 questionnaires kept for analysis. No statistically significant differences were found between the two groups of respondents. It was concluded that no significant non-response bias was evident in the data.

6.2 A FIRST LOOK AT THE DATA

This section presents the results of Phase III in a descriptive fashion. Doing so helps circumscribe the domain of this phase of the study, as well as it helps understand what UDA is for the organizations and users who participated to the research project. The descriptive statistics are presented so as to answer the following questions: What type of individual develops applications? What types of applications are developed? How are the applications developed and run? What are, from the user point of view, the advantages and disadvantages of UDA? How is user productivity affected by the applications developed? How do users evaluate their experience with UDA?

6.2.1 What Type of Individual Develops Applications?

General Background Information

Tables 6.5 and 6.6 provide some background information on the respondents (Questionnaire, Section V: Q2., Q3.) [1]. As indicated by Table 6.5, Accounting and Finance are the functional areas with the largest proportion of users. All firms have users in Accounting. Seventy three percent of the respondents of company E are from Accounting. This high percentage is explained by the fact that the main programming tool available to users at E, FORESIGHT, was introduced with the specific purpose of facilitating accountants' work [2]. Seventy five percent of respondents from J are either in Accounting (40%) or in Finance (35%). In this case as well, these percentages are related to the programming tools available to users. ASI-ST, a report generator, is mainly used by users with accounting related activities, while IFPS is designed for financial planning.

All firms except H have users in Finance, all except B have users in General Management, while all have users in Marketing. Engineering and Production users are concentrated in D and H (manufacturing) and in C, F, and I (utility). All Actuarial users are from B, the insurance company, while five of the seven Personnel users are from the financial institutions (A and G). The category Other includes departments such as Security, Economics, Operations Research, and so on. None of those functional areas includes more than 1.5% of the respondents.

TABLE 6.5

DISTRIBUTION OF RESPONDENTS PER FUNCTIONAL AREA

<u>FUNCTIONAL AREA</u>	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
ACCOUNTING	65	24.3
FINANCE	53	19.8
ENGINEERING	32	11.9
MARKETING	29	10.8
GENERAL MANAGEMENT	24	9.0
PRODUCTION	17	6.3
ACTUARIAL	11	4.1
PERSONNEL	7	2.6
OTHER	30	11.2

TOTAL	268	100%

NUMBER OF MISSING OBSERVATION: 4

TABLE 6.6

DISTRIBUTION OF RESPONDENTS PER JOB CATEGORY

<u>JOB CATEGORY</u>	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
ANALYST	77	29.0
SUPERVISOR or COORDINATOR	68	25.6
SYSTEMS ANALYST	23	8.6
MANAGER	20	7.5
ACCOUNTANT	16	6.0
ENGINEER	13	4.9
CLERK	13	4.9
PROGRAMMER	7	2.6
ACTUARY	4	1.5
SENIOR MANAGER	4	1.5
OTHER	21	7.9

TOTAL	266	100%

NUMBER OF MISSING OBSERVATIONS: 6

Table 6.6 outlines the distribution of respondents per job category. It should be noted that 11% of the respondents, namely Systems Analysts and Programmers, have "official" DP professional titles. Furthermore, the Analyst group includes some individuals who have titles such as Business Analyst or Accounting Analyst, but who have a response profile quite similar to the response profile of those DP professionals [3].

The category Supervisor or Coordinator, consists of people who have titles such as Budget Coordinator, Office Supervisor, and so on. However, some of them have responsibilities which appear to be more heavily oriented toward application development. Eighteen of the 68 supervisors (4.5% of all respondents) were found to have such responsibilities, with titles like Customer Systems Supervisor, Personnel Systems Supervisor, etc.

Four respondents were classified as Senior Managers; these respondents have job titles such as Vice-president, Controller, and so on. The 20 respondents falling in the category Manager are mostly managers in staff areas; they have titles such as Employee Benefits Manager, Manager Economic Studies, etc.

The User Taxonomy

Section 5.1 proposed a taxonomy of the population of users who develop applications. The taxonomy identified three main categories of respondents. First is the Micro DP Department category, consisting of users who typically develop applications requested by other users.

Second are the Staff Analysts for whom application development is a tool they use to assist them in solving problems. Finally, the Opportunity Seeker category is composed of more independent as well as more creative users. During the interviews the latter category was found to include a large majority of users who were either at the level of supervisor or manager, generally in a staff area.

The 272 questionnaire respondents were classified according to the taxonomy, because it was assumed that the presentation of the questionnaire results would be enriched by relating them to the three types of users rather than considering all users as belonging to the same homogeneous group.

The classification of the questionnaire respondents involved four main steps, detailed below. First, each questionnaire was examined and the respondent classified in one of the three groups. Second, discriminant analysis was performed in order to check for the consistency of the classification. Third, the questionnaires of those respondents for whom the first two steps did not agree were re-examined, and their category determined. Discriminant analysis was then performed again, as a final check.

During the first step, six variables were mainly used to classify the respondents. Table 6.7 lists these variables (Section II: Q2., Q11.B, Q12D.d., Q12E.e.; Section V: Q4.). When the questionnaires were examined to classify the respondents, the sixth variable, Job Title, was used only in order to "double-check" the classification.

TABLE 6.7

VARIABLES USED TO CLASSIFY RESPONDENTS

- (1) Percentage of the working hours which are spent to perform application development activities (Section II, Q1.)
- (2) Percentage of the applications a user developed, which were developed for somebody else or the person to whom the user reports (Section II, Q11 B.)
- (3) Number of months of experience with UDA (Section II, Q10.)
- (4) Percentage of the applications a user developed, which are run by the user's subordinates (Section II, Q12 D.)
- (5) Percentage of the information provided by the applications he developed, are used by the user to formulate a suggested action which is presented to the individual to whom the user reports (Section II, Q12 E.)
- (6) Job title (Section V, Q3.)

For instance, the fact that a user had Programmer as a job title confirmed that his classification in the Micro DP Department category was correct. Table 6.8 compares the three categories of users on the basis of the six variables used. As table 6.8 suggests, the classification of the respondents was based more on the comparative value of a variable rather than on its absolute value.

Micro DP Department. Those users who spent a larger percentage of their time developing applications and who developed a large percentage of applications for people other than themselves or the person to whom they reported generally fell in the Micro DP Department category. Job titles most common to this category are Programmer, Systems Analyst, Clerk, Supervisor of a group who has (from the user side) the

TABLE 6.8

COMPARISON OF THE THREE CATEGORIES OF USERS

<u>VARIABLE</u>	<u>MICRO DP DEPARTMENT</u>	<u>STAFF ANALYST</u>	<u>OPPORTUNITY SEEKER</u>
% OF HOURS	High	Low	Low
% DEVELOPED FOR OTHERS	High	Low	Low
MONTHS OF EXPERIENCE	Low to High	Medium	High
% RUN BY SUBORDINATES	Low	Low	Medium
% OF INFORMATION USED TO MAKE RECOMMENDATIONS	Low	High	Medium
JOB TITLES MOST OFTEN ENCOUNTERED	.Programmer .System Analyst .Clerk .Pseudo Analyst .Supervisor, System Group.	.Analyst .Financial Analyst .Engineer .Accountant .Actuary	.Senior manager .Manager .Supervisor .Senior Analyst

responsibility for a system, and "Pseudo-Analysts." The last job title was given at the time of coding the questionnaire to those users whose response pattern was very similar to the response pattern of Programmers or Systems Analysts, but who indicated the more general title of "Analyst."

Staff Analyst. The Staff Analyst category is made up of users who, typically, do not spend as much time performing UDA activities as users in the previous category. Moreover, while they sometimes develop

applications for users other than themselves or the person to whom they report, the percentage is generally lower than it is for users from the Micro DP Department group. On the basis of the variables used to classify users, the main distinctive characteristic of the Staff Analyst group is that they tend to use a higher percentage of the information provided by the applications they develop, in order to make recommendations to the person to whom they report. Staff Analysts were found to have job titles such as Analyst, Financial Analyst, Engineer, Accountant, Actuary.

Opportunity Seeker. Finally, the Opportunity Seeker category includes users who have a relatively long experience with UDA. Very often, the applications they develop are run by subordinates. In general, they do not spend as high a percentage of their working hours performing UDA activities, as Micro DP Department users may do. While they develop some applications for people other than themselves or the person to whom they report, the percentage of such applications is lower than it is for users in the Micro DP Department group. Such users have job titles like Senior Analyst, Supervisor, Coordinator, Manager, or Senior Manager.

The major concern raised by the classification process described above is one of consistency. While much effort was put on keeping a uniform "classification rule" throughout the 272 questionnaires, there were two major threats to consistency. First, the classification process required several hours, and researcher fatigue might have interfered with the uniformity of application of the rule. Second, a

number of questionnaires fell outside the bounds of the classification rule, and had to be dealt with on an individual, exception basis. In these cases, knowledge of UDA acquired during the interview phase had to be used. The issue, with regard to such exceptions, was: were all exceptions treated in the same fashion?

The second step of the classification process addressed these two issues. Discriminant analysis was performed on the questionnaire data, using SPSS [4]. Five of the six variables which were found most helpful in "manually" classifying respondents were used in the discriminant analysis. The variable Job Title, being a nominal variable, was excluded. Once the discriminant functions had been determined, SPSS provided the option of going back to each case, compute its discriminant score, and indicate the group to which the case has the highest probability of membership. The results of the classification are summarized in Table 6.9.

The percentage of correctly classified cases achieved, 71%, is quite satisfactory by conventional research standards. However, the questionnaires of the misclassified cases had to be checked, in order to determine the causes of the misclassifications. Seventy nine cases had been misclassified. For 22 of those 79 cases it was found that the classification rule had not been applied consistently with the other questionnaires. Interestingly, the 22 questionnaires were found to be mostly among the first or the last questionnaires to be classified. Investigator fatigue may explain the latter, while the former might be due to "lack of investigator experience." The classification of the

TABLE 6.9

RESULTS OF THE FIRST SPSS CLASSIFICATION OF USERS

ACTUAL GROUP	NUMBER OF CASES	PREDICTED GROUP MEMBERSHIP		
		1	2	3
GROUP 1 MICRO DP	106	83 78.3%	17 16.0%	6 5.7%
GROUP 2 STAFF ANALYST	90	17 18.9%	65 72.2%	8 8.9%
GROUP 3 OPP. SEEKER	71	5 7.0%	26 36.7%	40 56.3%
UNGROUPED CASES	5	1 20.0%	4 80.0%	0 0.0%

PERCENT OF "GROUPED" CASES CORRECTLY CLASSIFIED: . 70.4%

remaining 57 respondents was not changed.

Those 57 questionnaires were questionnaires which fell outside the bounds of the classification rule. First some had missing values for one or more of the discriminant variables. Since, in such a case, SPSS substitutes the total mean for the missing variable, some misclassifications occurred. Second, for the remaining questionnaires, investigator judgement was found to be more appropriate than the calculation of discriminant scores. For instance, one user (1) spent a very low percentage of his working hours performing application development activities; (2) did not develop applications for other people than himself or the person to whom he reported, (3) had little experience with UDA, (4) did not have applications run by subordinates,

and (5) used the information provided by some of the applications he had developed (15%) to make recommendations to his superior. This user was classified, by the discriminant analysis, as a Staff Analyst. However, the questionnaire indicated that the person transmitted directly to his superior the information provided by most of the applications (85%), and his job title was Accounting Clerk. This respondent actually belonged to the Micro DP Department category; yet, he played this role for one user only, his superior.

Finally, the fourth step in the classification process consisted of performing discriminant analysis a second time, in order to determine the improvement in the classification. Table 6.10 provides the results of this classification.

Table 6.11 shows the distribution of the respondents across the three groups. Note that the distribution of the questionnaire respondents among the three categories of the taxonomy is very close to the distribution of the 62 users interviewed (see Figure 5.5).

6.2.2 Characteristics of Users Who Develop Applications

Following the above discussion of the classification of respondents, the statistics pertaining to the characteristics of users who develop applications are presented below. This presentation will highlight the differences between the user categories, when such differences are statistically significant.

TABLE 6.10
RESULTS OF SPSS SECOND CLASSIFICATION

ACTUAL GROUP	NUMBER OF CASES	PREDICTED GROUP MEMBERSHIP		
		1	2	3
GROUP 1 MICRO DP	106	88 83.0%	14 13.2%	4 3.8%
GROUP 2 STAFF ANALYST	90	11 12.2%	71 78.9%	8 8.9%
GROUP 3 OPP. SEEKER	71	4 5.6%	23 32.4%	44 62.0%
UNGROUPED CASES	5	0 0.0%	5 100.0%	0 0.0%

PERCENT OF "GROUPED" CASES CORRECTLY CLASSIFIED: 76.03%

TABLE 6.11
DISTRIBUTION OF RESPONDENTS PER TAXONOMY CATEGORY

<u>CATEGORY</u>	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
MICRO DP DEPARTMENT	106	39.7
STAFF ANALYST	90	33.7
OPPORTUNITY SEEKER	71	26.6
	-----	-----
	267	100%

NUMBER OF MISSING OBSERVATIONS: 5

Three statistical tests were used to determine the significance of the differences between the three categories of respondents. First, analysis of variance was performed for variables that had been measured on an interval or ratio scale. Furthermore, an a posteriori test, Scheffe's test, was performed. This test compares all possible pairs of group means, in order to determine the ones that are different. The alpha chosen for Scheffe's test is .05. Kruskal-Wallis one way analysis of variance was performed on those variables which were measured on an ordinal scale, while a Chi-Square test was used for nominal variables [5].

Table 6.12 summarizes the results of the analyses performed on variables related to the question: Who develop applications? Once put together, those variables help draw a profile of users for each category. Table 6.12 should be read as follows. For all the variables, the null hypothesis - that the three groups are from the same population - was rejected. In all cases, the level of significance was more stringent than .004 (Tables 6.13 to 6.15 and Figure 6.4 provide details on each analysis). The arrows drawn between two categories of the taxonomy indicate those groups between which a significant difference exists, on a one-on-one basis. All the results of the a posteriori tests were significant at levels more stringent than .05.

For instance, there is a significant difference, among the three groups, in the average percentage of working hours spent performing UDA activities. In particular, the a posteriori test indicates that the

TABLE 6.12

SUMMARY OF ANALYSES FOR: WHO DEVELOPS APPLICATIONS

	MICRO DP DEPARTMENT	STAFF ANALYST	OPPORTUNITY SEEKER
% OF HOURS	43.5%	18.4%	17.0%
% DEVELOPED FOR OTHERS	67.9%	17.2%	17.9%
USER DEVELOPS APPLICATIONS FOR OTHERS	89.5%	48.2%	56.3%
FORMAL DUTY TO DEVELOP APPLICATIONS FOR OTHERS	80.9%	55.0%	51.3%
GROUP WORK WITH INFORMATION	8.8%	21.4%	27.8%
USES INFORMATION PERSONALLY	7.0%	22.0%	21.0%
TRANSMITS INFORMA- TION TO OTHER USERS	56.0%	21.3%	23.8%
MAKE RECOMMEN- DATIONS	5.3%	19.5%	12.1%
MONTHS OF EXPERIENCE	33	24	60
NUMBER OF APPLICA- TIONS DEVELOPED	34	13	62
PREVIOUS JOB	70.8%	76.7%	38.6%

difference is significant between Micro DP Department and Opportunity Seekers as well as between Micro DP and Staff Analysts. However, the a posteriori test shows no significant difference between Staff Analysts and Opportunity Seekers.

For eight variables out of the 11 listed in Table 6.12, Micro DP Department users have responses which significantly differ from each of the other two groups. For two other variables (Make Recommendations and Previous Job) the Micro DP Department category is different from at least one other category. On the other hand, Staff Analyst and Opportunity Seeker categories have more affinities, with respect to the variables listed in Table 6.12. They significantly differ on three variables only: the number of months of experience with UDA, the number of applications they have developed, and the job prior to beginning application development. The results of the analyses, which will be discussed below, suggest that the category Opportunity Seeker might be a natural "next career step" for the Staff Analyst category.

Table 6.13 summarizes the results of analysis of variance performed on four of the variables of Table 6.12. These variables are the percentage of work hours spent performing UDA activities (Section II: Q1.), the percentage of applications developed for other users than the user himself or his superior (Section II: Q11B.), the number of months of experience with UDA (Section II: Q9.), and the number of applications developed (Section II: Q10.). The first three variables were used in the discriminant analysis. As for Table 6.12, an arrow drawn between two groups indicates a statistically significant

TABLE 6.13
COMPARISONS ACROSS USER CATEGORIES

	<u>MICRO DP DEPARTMENT</u>	<u>STAFF ANALYST</u>	<u>OPPORTUNITY SEEKER</u>	<u>ANOVA SIGNIFICANCE</u>
% OF HOURS	43.5%	18.4%	17.0%	.0001
% DEVELOPED FOR OTHERS	67.9%	17.2%	17.9%	.0001
MONTHS OF EXPERIENCE	33	24	59	.0001
NUMBER OF APPLICATIONS	33	13	61	.004

difference between these groups. For the variables in Table 6.13, the Scheffe's test was significant at .05 .

These results suggest that Micro DP Department users spend a larger percentage of their working hours developing applications, and that they develop a larger percentage of applications for other users than for themselves or for the person to whom they report. Opportunity Seekers are the users who have the longest experience with UDA and have developed the largest number of applications 6 .

Users were also asked if they developed applications for people other than themselves or the person to whom they report (Section II: Q11D.). In the case of a positive answer, users were further asked if developing applications for others was a formal duty (Section II:

TABLE 6.14

USER CATEGORY VS. APPLICATION DEVELOPMENT FOR OTHERS

DEVELOP APPLICATIONS FOR OTHERS?

	COUNT	I	YES	NO	ROW TOTAL
ROW PCT	I				
COL PCT	I				
TOT PCT	I				
	I	I	I	I	I
	I	I	I	I	I
MICRO DP DEPARTMENT	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
STAFF ANALYST	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
OPPORTUNITY SEEKER	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
	I	I	I	I	I
COLUMN TOTAL					

CHI SQUARE= 41.31117 WITH 2 DEGREES OF FREEDOM

SIGNIFICANCE = 0.0000

NUMBER OF MISSING OBSERVATIONS = 11

Q11C.). Tables 6.14 and 6.15 present the results of a Chi-square test for each question.

Table 6.14 indicates that while a majority of respondents do develop applications for others, the percentage of Micro DP Department type of users who do so is larger than it is for the other two categories. An a posteriori test was performed, and the results

TABLE 6.15

USER CATEGORY VS. FORMAL DUTY

	FORMAL DUTY TO DEVELOP APPLICATIONS FOR OTHERS?				ROW TOTAL
	COUNT	YES	NO		
	RGW PCT				
	COL PCT				
	TOT PCT				
		I	I	I	I
		I	I	I	I
MICRO DP DEPARTMENT		I 76	I 18	I	I 94
		I 80.9	I 19.1	I	I 54.3
		I 64.4	I 32.7	I	
		I 43.9	I 10.4	I	
		I	I	I	I
STAFF ANALYST		I 22	I 18	I	I 40
		I 55.0	I 45.0	I	I 23.1
		I 18.6	I 32.7	I	
		I 12.7	I 10.4	I	
		I	I	I	I
OPPORTUNITY SEEKER		I 20	I 19	I	I 39
		I 51.3	I 48.7	I	I 22.5
		I 16.9	I 34.5	I	
		I 11.6	I 11.0	I	
		I	I	I	I
COLUMN TOTAL		118	55		173
		68.2	31.8		100.0

CHI SQUARE = 15.29962 WITH 2 DEGREES OF FREEDOM

SIGNIFICANCE = 0.0005

NUMBER OF MISSING OBSERVATIONS = 99

indicate that this percentage is also significantly different from each one of the other two categories. The result of the a posteriori test is significant at a more stringent level than .0001.

Moreover, as indicated by Table 6.15, among those users who develop applications for people other than themselves or the person to whom they report, the percentage of Micro DP Department users for whom doing so is a formal duty (80.9%) is larger than for the two other

groups. In this case as well, an a posteriori test confirms that a statistically significant difference exists between Micro DP Department and each one of the other two user category. This test was significant at the .005 level.

These results are consistent with the observation, previously mentioned, that the percentage of applications Micro DP Department type of users develop for others is larger than for the other two categories of users.

Table 6.16 compares users with respect to job they had at the time of starting to develop applications (Section II: Q7.). The results support the suggestion made earlier that the category Opportunity Seeker is a natural "next career step" for Staff Analysts. While 76.7% of the latter category had the same job as they have now when they started developing applications, only 38.6% of the Opportunity Seekers did. As previously mentioned, Opportunity Seekers are mainly found among coordinators, supervisors, and managers, typically in staff functions such as financial planning. The questionnaire responses suggest that their previous job was of the analyst type. It may be suggested that as the Staff Analyst user evolves in his career, his profile with respect to UDA, evolves toward that of Opportunity Seeker.

The questionnaire asked users to indicate the use which was made of the information provided by the applications they had developed (Section II: Q12E.). Figure 6.1 should be interpreted as follows. Respondents were presented with six categories of information usage:

TABLE 6.16
USER CATEGORY VS. JOB PRIOR TO UDA

	COUNT	I			
	ROW PCT	I	YES	NO	ROW
	COL PCT	I			TOTAL
	TOT PCT	I			
		I			I
		I	75	I 31	I 106
MICRO DP		I	70.8	I 29.2	I 39.8
DEPARTMENT		I	43.9	I 32.6	I
		I	28.2	I 11.7	I
		I	69	I 21	I 90
STAFF ANALYST		I	76.7	I 23.3	I 33.8
		I	40.4	I 22.1	I
		I	25.9	I 7.9	I
		I	27	I 43	I 70
OPPORTUNITY		I	38.6	I 61.4	I 26.3
SEEKER		I	15.8	I 45.3	I
		I	10.2	I 16.2	I
COLUMN			171	95	266
TOTAL			64.3	35.7	100.0

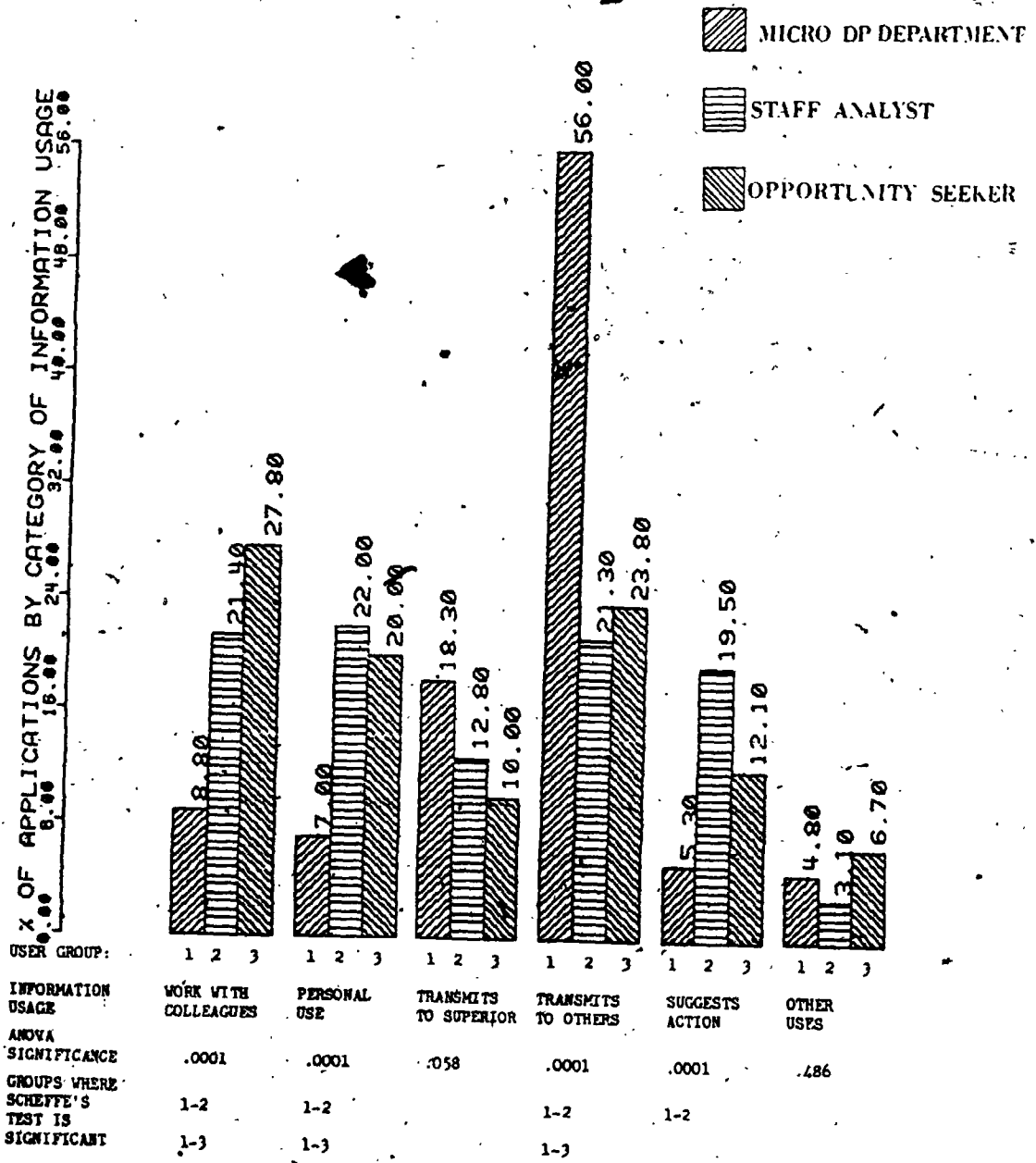
CHI SQUARE = 28.10096 WITH 2 DEGREES OF FREEDOM
SIGNIFICANCE= 0.0001

NUMBER OF MISSING OBSERVATIONS = 6

(1) use the information as a basis for group work with colleagues, (2) personally use the information, (3) directly transmit the information to superior, (4) directly transmit the information to other users, (5) use the information to make recommendations, and (6) other uses. Users were asked to indicate which percentage of the applications they had developed fell in each category.

Figure 6.1 presents the average percentage for each category of

FIGURE 6.1
INFORMATION USAGE



X OF APPLICATIONS BY CATEGORY OF INFORMATION USAGE

0.00 6.00 16.00 24.00 32.00 40.00 48.00 56.00

USER GROUP:
1 2 3

INFORMATION USAGE
ANOVA SIGNIFICANCE
GROUPS WHERE SCHEFFE'S TEST IS SIGNIFICANT

Category	ANOVA SIGNIFICANCE	GROUPS WHERE SCHEFFE'S TEST IS SIGNIFICANT
WORK WITH COLLEAGUES	.0001	1-2
PERSONAL USE	.0001	1-2
TRANSMITS TO SUPERIOR	.058	1-3
TRANSMITS TO OTHERS	.0001	1-2
SUGGESTS ACTION	.0001	1-2
OTHER USES	.786	

information usage, and compares the three groups of respondents. It includes the level of significance of the analysis of variance and it also indicates the groups between which Scheffe's test was significant.

As for the other variables, Micro DP Department users are significantly different from each of the other two groups. The most common use of the information provided by the applications they develop is to directly transmit it to other users (18.3% to their superior and 56% to users other than their superior). While the other categories of users directly transmit information as well (32.1% for Staff Analyst and 33.8% for Opportunity Seekers), they typically have a more active role in the use of information.

The last aspect of UDA which is to be used to answer the question of who develops applications is the computer background of the users (Section II: Q8.). Kruskal-Wallis one way analysis of variance was performed and the results indicate that there is no significant difference among the three categories of respondents. Table 6.17 shows the distribution of the variable computer background. These results indicate that a majority of respondents had at least some knowledge of computers prior to beginning their UDA activities.

To sum up, the questionnaire respondents were classified according to the user taxonomy which had been developed during the interview phase of the study. Micro DP Department users significantly differ from the two other categories of users, on variables such as proportion of working hours spent on UDA activities, percentage of applications

TABLE 6.17
COMPUTER BACKGROUND OF USERS

	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
ABSOLUTELY NO KNOWLEDGE OF COMPUTERS	36	13.2
SOME USE OF COMPUTERS	32	11.8
SOME PROGRAMMING	132	48.5
GOOD KNOWLEDGE	72	26.5
	-----	-----
	272	100.0%

NUMBER OF MISSING OBSERVATIONS: 0

which are developed for other people, the usage made of the information provided by the applications developed, and so on.

6.2.3 Characteristics of Applications Developed by Users

Five characteristics of the applications users develop are examined in this section. These dimensions are the functions performed by the applications, the organizational entities the applications support, the length of time required to develop the applications, the source of the data used in the applications, and finally, the periodicity of use of the applications.

Functions Performed

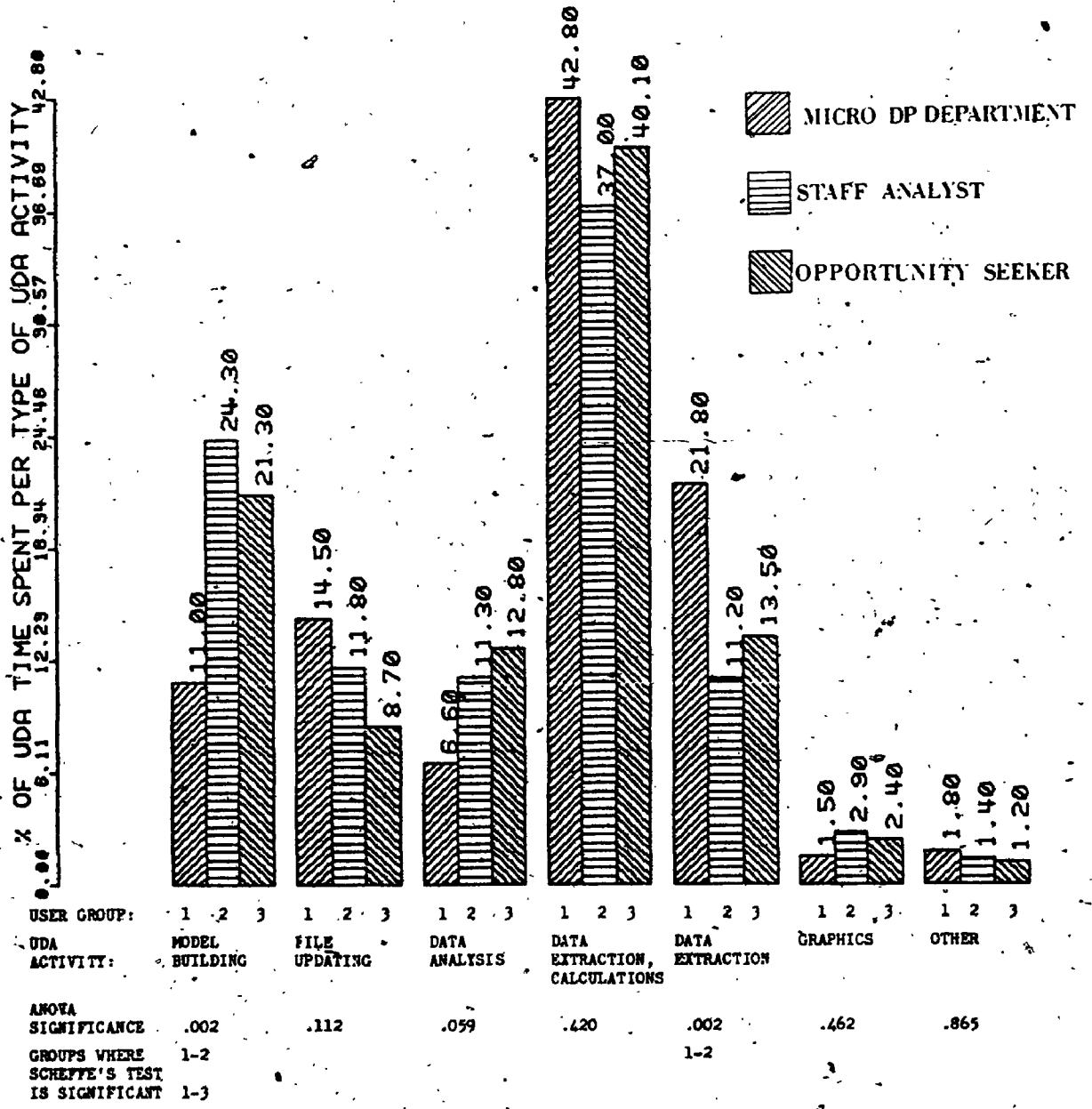
Figure 6.2 illustrates the distribution of the time spent on UDA across UDA activities (Section II: Q2.). Those activities, which are based on the application taxonomy presented in Chapter V, are: (1) creating models, (2) creating programs which update files, (3) creating programs which perform data analysis, (4) creating programs which extract data, perform calculations on the data, and produce reports, (5) preparing programs which extract data and produce reports, and (6) preparing graphics. For all three categories of the user taxonomy, the activity which receives the largest proportion of UDA time is the development of applications which extract data, perform calculations on data, and produce a report. There is no statistically significant difference between the three groups of users, with respect to the percentage of time spent on this activity.

However, the results of analysis of variance indicate a significant difference in the percentage of users' time spent on model building and on extraction of data. The results suggest that Micro DP Department users spend a smaller proportion of their UDA time developing models than the other two groups do. On the other hand, the Micro DP Department spend a larger proportion of the time creating programs which extract data only, without performing any calculations.

For the three groups, the activity which receives the smallest average percentage of time is the preparation of graphics. This may be explained by the fact that the preparation of graphics may be accessory to other UDA activities. Also, as pointed out in Chapter V, not all firms provided users with graphics capacities, and those who did were just introducing them. The categories "Other" is small and

FIGURE 6.2

PERCENTAGE OF UDA TIME SPENT PER UDA ACTIVITY



includes activities such as "preparing tapes to send to a service bureau."

Organizational Entities Supported

No statistically significant difference was found among the three categories of users, with regard to the organizational entities supported by the applications developed (Section II: Q3.). As suggested by Table 6.18, organizational support applications is the category which receives the smallest percentage of users' UDA time.

TABLE 6.18

PERCENTAGE OF UDA TIME PER ORGANIZATIONAL ENTITY SUPPORTED

	<u>AVERAGE PERCENTAGE</u>
INDIVIDUAL SUPPORT APPLICATIONS	37.2
GROUP SUPPORT APPLICATIONS	41.1
ORGANIZATIONAL SUPPORT APPLICATIONS	21.7

	100.0%

NUMBER OF MISSING OBSERVATIONS: 9

This result supports McLean's (1979) claim that UDA efforts would (and should) be oriented more toward individual or group support applications rather than toward organizational support applications. However, the results indicate that even though their percentage is

low, the latter applications are indeed developed by users. Both interviews and questionnaires suggest that such applications typically are planning and budgeting type of applications, where indeed several departments may be involved. Group support and individual support applications receive approximately similar percentages of UDA time by users.

Time Required for Development

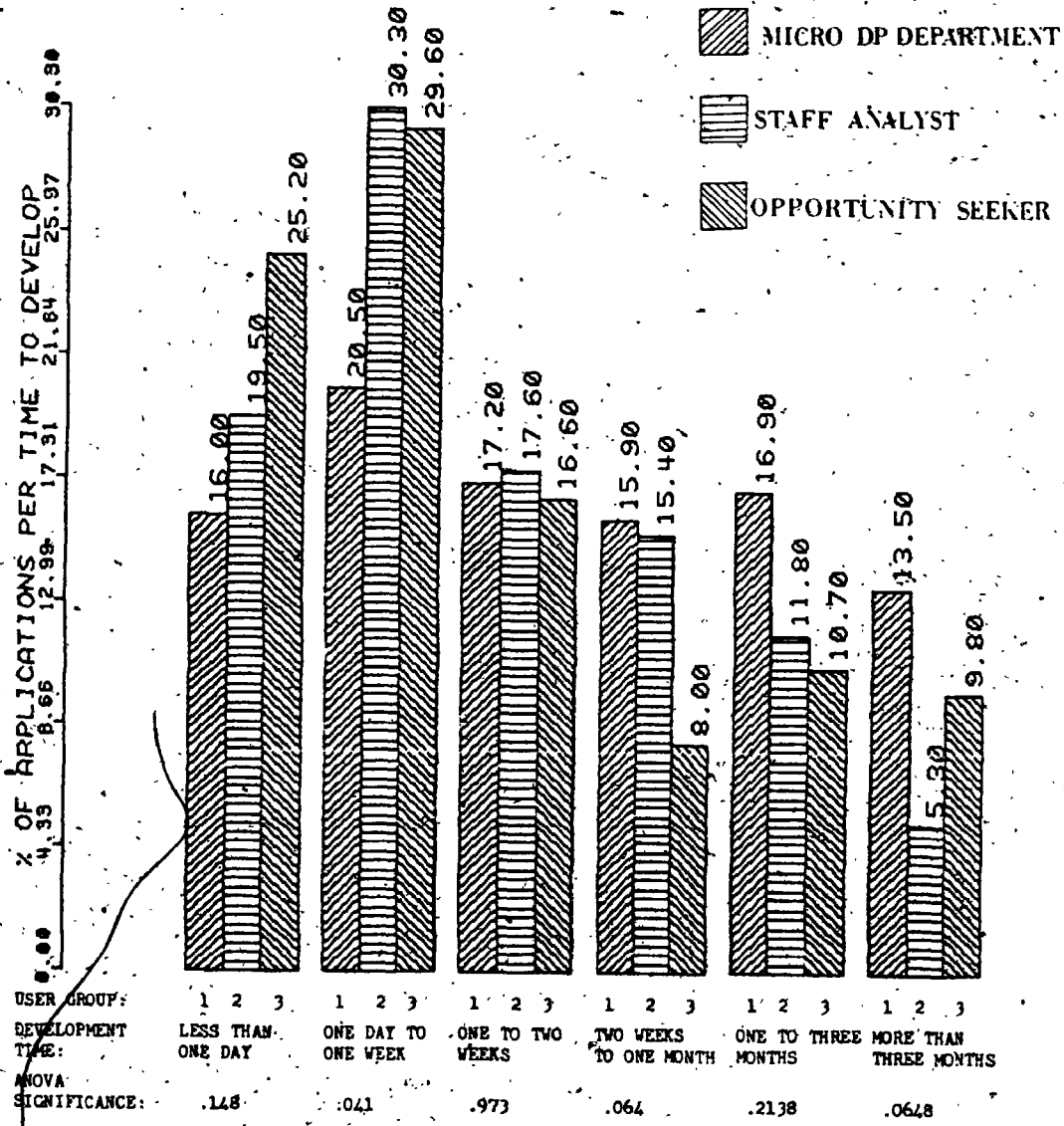
Figure 6.3 compares the three categories of users with respect to the time required to develop applications (Section II: Q12A.). In general, it can be said that the applications developed by users are "small" applications, by DP standards. For the three groups, only a small percentage of applications took more than three months to develop. For the Staff Analyst and Opportunity Seeker users, the mode is in the range of "one day to one week," while in the case of Micro DP Department users, an almost equal percentage of applications fall in each class.

Source of Data

For both Staff Analyst and Opportunity Seeker categories, 58% of the applications developed use files created by inputting data from documents (see Figure 6.4; Questionnaire Section II: Q12B.). In the case of Micro DP Department users the majority of applications use files made available by DP. These results are consistent with the type of functions performed by the applications (see Figure 6.2). As discussed earlier, Micro DP Department users spend more time developing applications which simply extract data. The results

FIGURE 6.3

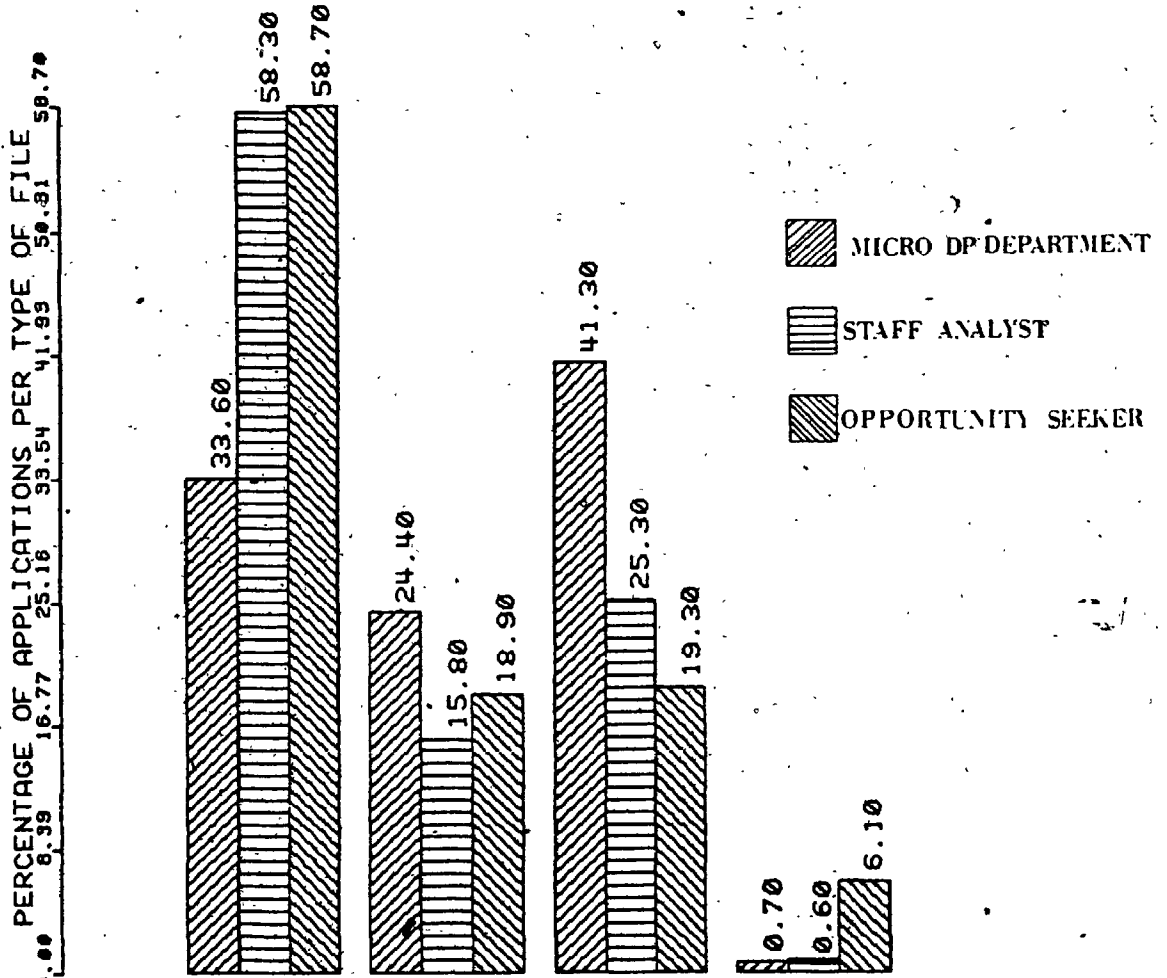
TIME REQUIRED FOR DEVELOPMENT



SCHIFFE'S TEST NOT SIGNIFICANT

FIGURE 6.4

SOURCE OF DATA FOR APPLICATIONS



ANOVA SIGNIFICANCE:	GROUPS WHERE SCHEFFE'S TEST IS SIGNIFICANT
.0001	1-2
.0957	1-3
.0002	1-2
.0024	1-3

suggest that for such extractions, DP files are used directly. On the other hand, Staff Analysts and Opportunity Seekers spend a relatively large percentage of their time developing models. During the interviews of Phase II it was found that model-type applications generally use files created from documents rather than using DP files.

Periodicity of Use of Applications

Finally, as indicated in Table 6.19, most applications (66.8%) are run on a regular basis (Section II: Q12C.). This observation is at variance with what is often said about applications which users develop (or should develop), that is that they should be "one-shot efforts, mainly consisting of throwaway code." However, other research (Rockart and Flannery, 1981), showed that users actually developed an important proportion of regularly run applications.

In Table 6.19, the category Other mainly consists of applications which are run on demand (such as loan analysis applications), and of applications which are run once a year (budgeting applications, for instance).

6.2.4 How Are the Applications Developed and Run?

Tables 6.20 to 6.23 and Figure 6.5 provide information on how the applications are developed and run.

TABLE 6.19
 PERIODICITY OF USE OF APPLICATIONS

	<u>AVERAGE PERCENTAGE</u>
ONE SHOT APPLICATIONS	24.4%
REGULARLY RUN APPLICATIONS	66.8%
OTHER	8.8%
	<hr style="width: 10%; margin-left: auto; margin-right: 0;"/>
	100.0%

NUMBER OF MISSING OBSERVATIONS: 12

Application Development Mode

Table 6.20 suggests that users typically develop applications alone; on the average, the percentage of UDA activities that users perform alone is nearly 75%. The category "Other" includes modes such as assisting other users, supervising other users, and working with the IC consultants (Section II: Q4.).

Computer Used for UDA

On the average, 85% of the respondents' UDA activities are performed using their firm's central computer (Table 6.21; Questionnaire Section II: Q5.). Such a high percentage was predictable, since users who received a questionnaire had an "account number" on the main computer. Consequently, this percentage might be slightly higher than the actual percentage of UDA activities that use the mainframe in the firms studied. For those users who responded to

TABLE 6.20

APPLICATION DEVELOPMENT MODE

	<u>AVERAGE PERCENTAGE</u>
ALONE	73.1%
IN USER TEAMS	24.3%
OTHER	2.6%
	<hr/>
	100.0%

NUMBER OF MISSING OBSERVATIONS: 6

TABLE 6.21

AVERAGE PERCENTAGE OF UDA ACTIVITIES PER COMPUTER TYPE

	<u>AVERAGE PERCENTAGE</u>
COMPANY'S CENTRAL COMPUTER	85.2%
ANOTHER COMPANY'S COMPUTER	5.8%
DEPARTMENT'S COMPUTER	5.5%
PERSONAL COMPUTER	0.9%
OTHER	2.6%
	<hr/>
	100.0%

NUMBER OF MISSING OBSERVATIONS: 5

the questionnaire, very little use is made of personal computers. Such a result is consistent with information obtained during the interview phase, when DP departments were aware of only a small number of micro computers in use in their organizations. However, some DP managers commented that they felt that users were aware of the possibilities of such computers and that their number might very well increase.

Terminal Used for UDA

The majority (58%) of the hours users spend using a terminal are spent on a terminal the user shares with others in his department (see Table 6.22; Questionnaire Section II: Q6A.). Twenty three percent of the terminal hours are spent using a dedicated terminal in the office, while 10% of the hours are spent using a terminal located in an IC. Somewhat surprisingly, fully 213 respondents (79% of the sample) indicated that they never use an IC terminal, while 24 respondents (7.8%) spend more than 80% of their terminal hours using an IC terminal. Finally, 4% of the terminal hours are spent using a terminal located in DP but not in an IC, 3% are spent use a home terminal, and 2.7% using another mode of running applications. The latter consists mainly of the hours spent using punched cards. Twenty six percent of the respondents indicated that they sometimes use punched cards or that they fill out forms which are transmitted to DP (Section II: Q6B.).

TABLE 6.22

AVERAGE PERCENTAGE OF UDA ACTIVITIES PER TERMINAL LOCATION

	<u>AVERAGE PERCENTAGE</u>
USE A DEDICATED TERMINAL	23.9%
SHARE TERMINAL(S) WITH OTHERS IN DEPARTMENT	58.4%
USE AN IC TERMINAL	10.7%
USE A DP TERMINAL (OTHER THAN IC)	4.2%
USE A HOME TERMINAL	0.3%
OTHER	2.5%
	<hr/> 100.0%

NUMBER OF MISSING OBSERVATIONS: 6

Programming Tools

The programming tools are used to develop and run applications is an important aspect of how applications are developed and run. Fully 98 different programming tools were mentioned by respondents as being the tool they most often use (Section IV: Q1A.). However, as indicated by Table 6.23, 12 of these tools account for 82% of the responses. The category "Other" includes tools such as FPS, EXTRACTO, PASCAL, SUPERWYLBUR, MARK IV, and so on. None of those tools is used by more than 2% of the respondents. Except for some tools in the category "Other", all tools were made available by DP. Among the

TABLE 6.23
PROGRAMMING TOOLS USED

	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
GIS	36	13.2
APL	27	10.0
ADRS II	25	9.3
SAS	25	9.3
EASYTRIEVE	25	9.3
IFPS	16	5.9
FORTRAN	16	5.9
BASIC	12	4.5
GENMOD	12	4.5
FORESIGHT	9	3.3
PLANCODE	7	2.6
OTHER	49	18.5
	-----	-----
	269	100.0%

NUMBER OF MISSING OBSERVATIONS: 3

tools in the category "Other," some tools, such as GE.FALL II, RATEMOD, etc., are used on outside timesharing services.

To the question of which tools were used for a specific UDA activity, two noticeable patterns of responses occurred. First, several users indicated that they used the same tool to develop different types of applications. Second, others indicated that for the same UDA activity, they would use several tools. The interview phase of the study provided some explanation for these two observations. The former type of response is related to the fact that

users generally learn how to use one tool only, or become a "fan" of one tool. In such a case, they use the same tool for several types of activities, even if the tool is not well suited for such activities. These users seem to be reluctant to adopt another tool. On the other hand, some users learn to use several tools and try to make the best use of each. For instance, some users interviewed at G used IFPS to develop models but preferred the output formatting of ADRS-II. They then wrote an APL routine to interface between their IFPS files and ADRS-II.

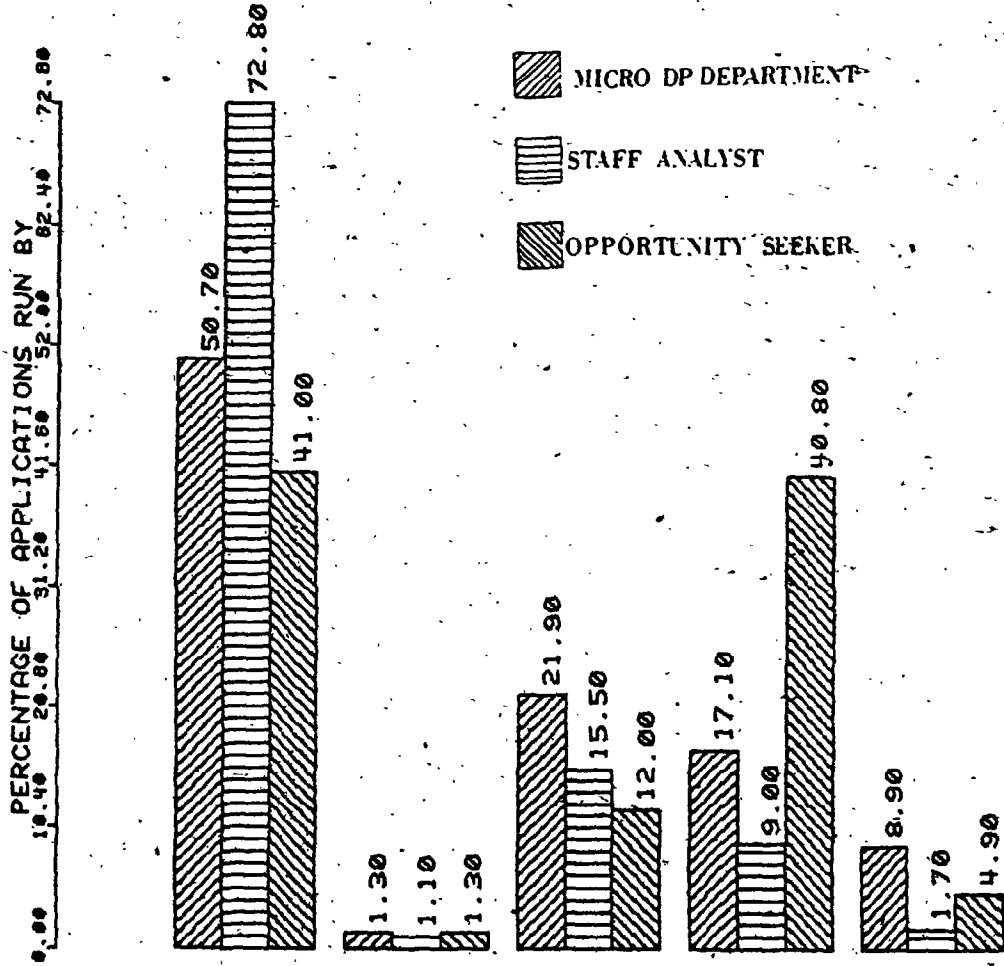
These two phenomena are related to the lack of integration of programming tools discussed in Chapter II. It is also related to the fact that some users are reluctant to learn new tools, either because they lack confidence in their computer background, or because such a learning would require extra efforts, or simply because they do not like change.

Who Runs the Applications?

While there was no statistically significant difference between the three categories of the taxonomy with regard to the above aspects of UDA, there were such differences with respect to the question: Who runs the applications? Figure 6.5 illustrates the responses obtained to this question (Section II: Q12D.). A first interesting result is the low percentage of applications (slightly over 1%) which are run by the respondents' superior. This result is consistent with the high average percentage of applications which are run by subordinates, in the case of the Opportunity Seeker category. As discussed previously,

FIGURE 6.5

WHO RUNS THE APPLICATIONS



PERSON WHO RUNS APPLICATIONS:	1	2	3
ANOVA SIGNIFICANCE:	.0001	.948	.0655
GROUPS WHERE SCHEFFE'S TEST IS SIGNIFICANT	1-2	1-2	1-3

these users often have managerial responsibilities. Forty percent of the applications they developed are run by subordinates. This suggests a tendency to delegate the task of running applications to subordinates. The low percentage of applications developed by users' superiors appears to be consistent with this finding. Both Micro DP Department users and Staff Analyst users run the majority of their applications themselves, while Opportunity Seekers indicate an equal percentage of applications run by subordinates. A significant difference exists, between user groups, with respect to the percentage of applications which fall in the category "Other." This category mostly consists of applications which are run by people in other departments, or as put by some respondents, "by the users for whom the applications were developed."

6.2.4 Advantages and Disadvantages of UDA

Respondents were asked what was, in their opinion, the major advantage and the major disadvantage of developing applications themselves rather than having DP do the work (Section II: Q14., Q15.). Tables 6.24 and 6.25 list the results obtained.

Advantages

The timeliness with which applications are developed was cited by 50.6% of the respondents as one major UDA advantage. Timeliness itself was mentioned by 31.6% of the respondents while it was mentioned together with other disadvantages by 19%. Such a result is not too surprising, since in most of the firms studied, DP faces a

TABLE 6.24
MAJOR ADVANTAGES OF UDA

	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
TIMELINESS	83	31.6
TIMELINESS & FLEXIBILITY	30	11.4
TIMELINESS & COST REDUCTION	14	5.3
TIMELINESS & NO COMMUNICATION PROBLEMS	6	2.3
FLEXIBILITY	17	6.5
COST REDUCTION	4	1.5
NO COMMUNICATION PROBLEMS	18	6.8
USE OF OWN EXPERTISE	55	20.9
INDEPENDENCE AND BETTER CONTROL OVER DEVELOPMENT PROCESS	20	7.6
OTHER	16	6.1
	-----	-----
	263	100.0%

NUMBER OF MISSING OBSERVATIONS: 9

large backlog of application development projects. In such cases, users have to wait substantial periods of time before the applications they request are completed. Moreover, some requests are given such a low priority that they might never be developed. For users in such situations, UDA's major advantage is likely to be timeliness. As some users explained during the interviews and on the questionnaire, it may

very well take them more hours of actual work to get an application developed, since they do not have the expertise of DP professionals. However, the applications do get developed much sooner.

The ability for users to use their own expertise, in order to arrive at an application which does exactly what it should do, was the second most often mentioned advantage. This advantage is somewhat related to two other advantages, namely, no communication problems and independence. These three advantages pertain to the idea that UDA frees users from the need to interact with DP, hence eliminates problems stemming from such interactions.

Disadvantages

However, UDA is not without inconveniences. While 10.5% of the respondents indicated that they perceived no disadvantage to UDA, 33.1% of the users stated that their lack of expertise with programming and the use of computers was a major disadvantage. This lack of expertise causes users to err for a while before finding, if not the best way to do something, at least a way that worked. Nineteen percent of the respondents indicated that UDA was too time consuming. Some users explain that UDA very often infringes on their "normal" duties. On the other hand some users (5.1%) claim that they do not have enough time to spend on application development, because they have other duties. It appears that these two disadvantages are two ways of viewing the same problem. Other disadvantages are somewhat related to users' lack of expertise. For instance, 7.0% of the respondents indicated that the programs they write are not as

TABLE 6.25
MAJOR DISADVANTAGES OF UDA

	<u>NUMBER OF RESPONDENTS</u>	<u>PERCENTAGE</u>
NO DISADVANTAGES	27	10.5
LACK OF EXPERTISE	85	33.1
TIME CONSUMING	50	19.5
SECURITY AND VALIDITY ISSUES	20	7.8
LACK OF EFFICIENCY OF CODE	18	7.0
DP RELATED DISADVANTAGES	17	6.8
LACK OF TIME TO DEVOTE TO UDA	13	5.1
OTHER	27	10.2
	-----	-----
NUMBER OF MISSING OBSERVATIONS:	15	5.8
	257	100.0%

efficient as they would be if they were written by DP professionals.

Some users indicated that they do not document their applications and that they are the only ones who know how the application works. This and other security and validity issues, such as the lack of audit trails, were mentioned by 7.8% of the respondents.

DP related disadvantages include things such as: DP does not provide all the support it should, the response time is poor, DP is reluctant to see users develop applications, DP does not provide enough information, etc. A total of 6.8% of respondents mentioned DP related disadvantages.

Finally, the category "Other" includes: the lack of resources in a given department to reap all the benefits of UDA, the perception, by top management, that UDA users are technicians because they know how to program and actually do programming, and so on.

6.2.5 Productivity Improvements

The questionnaire attempted to measure productivity improvements due to the applications users developed (Section II: Q16., Q17., Q18.). Two categories of applications were examined, namely one-shot applications and applications which are run on a regular basis. For one-shot applications, users were asked to compare the time it took to develop and run the application to the time it used to take or would have taken to perform the same task otherwise. The question was asked for both the one-shot applications which took the longest period of time to develop and the one which took the least time to develop. Users were also asked to compare the time it takes to run an average regularly run application, including the time to enter the data, to the time it used to take or would have taken using other means.

Table 6.26 summarizes the results [7]. For the purposes of this study, productivity factor is defined as "how much longer it used to take (or would have taken) to perform the same task without a given application." It should be noted that the results are somewhat conservative. Users were asked to check the number which represented their response. The numbers went up to 8. Some respondents indicated that the response was more than 8. In such case, the response was

TABLE 6.26
PRODUCTIVITY IMPROVEMENT

	<u>PRODUCTIVITY FACTOR</u>
SHORTEST ONE SHOT VALID CASES: 135 MISSING CASES: 137	4.93
LONGEST ONE SHOT VALID CASES: 146 MISSING CASES: 126	6.23
REGULARLY RUNN APPLICATION VALID CASES: 192 MISSING CASES: 80	6.16

coded 9, even if the respondent indicated "several times more." The results obtained in the present study are consistent with results reported by Rosenberger (1981b):

Such results strongly suggest that there are indeed productivity improvement aspects in UDA.

6.2.6 Users' Evaluation of UDA

The questionnaire asked the respondent to indicate, on a 5-point scale, the importance of 18 different aspects of UDA (Section III: Q1.). Later in the questionnaire they were asked to indicate the extent to which they were satisfied or dissatisfied with UDA, along the same dimensions (Section III: Q5.). These aspects are listed in Table 6.27.

TABLE 6.27

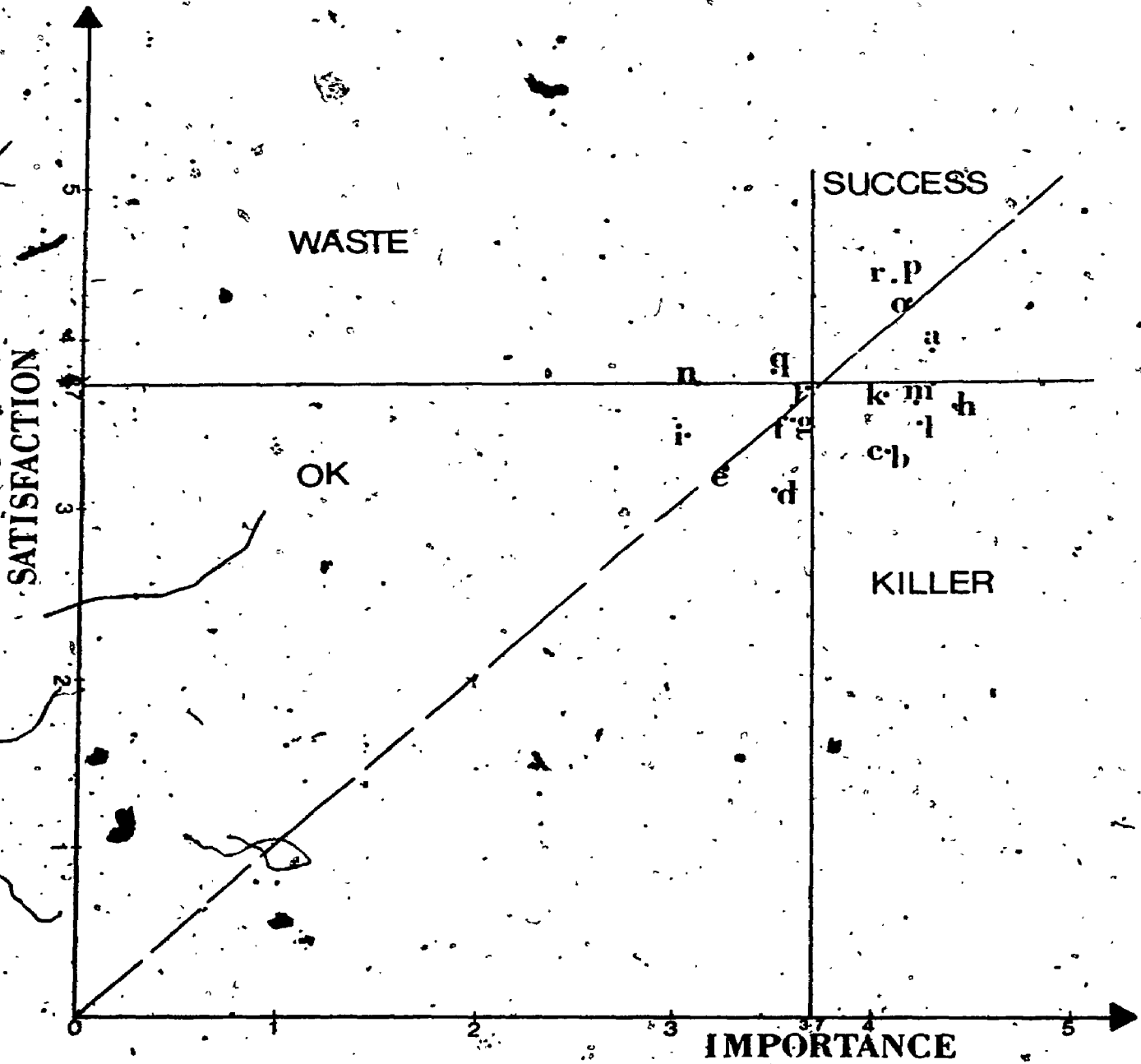
ASPECTS OF UDA EVALUATED BY USERS

- A. Ease of use of programming tools
- B. Awareness of the programming tools available
- C. Quality of training with the use of the tools
- D. Assistance in the choice of which tool to use
- E. Assistance during the design of applications
- F. Debugging assistance
- G. Assistance with technical aspects
- H. Consultant's knowledge of programming tools
- I. Consultants' understanding of the applications
- J. People orientation of consultants
- K. Availability of consultants
- L. Computer response time
- M. Availability of terminals
- N. No need to communicate application requirements to a DP professional
- O. Timeliness
- P. Control over applications
- Q. Cost reductions
- R. Usefulness of the information provided by applications

In order to determine if there is a fit between importance and satisfaction of the 18 UDA aspects, the results obtained are presented in Figure 6.6. using a technique suggested by Alloway (1980a). Importance and Satisfaction each occupy one axis, and the average score for each UDA aspects is plotted [8].

A first observation is that users are generally satisfied with UDA. None of the average satisfaction scores is lower than 3.1, on the 5 point scale. However, the graph suggests that there is a misfit between importance and satisfaction. A perfect fit would exist if all the data points were on the diagonal; yet, they are relatively scattered. Following Alloway's mode of presentation, the graph is

FIGURE 6.6
IMPORTANCE VS. SATISFACTION



divided into four quadrants. The location of the axes forming the quadrants is arbitrary. In the present situation, since the distributions are shifted toward high satisfaction, a "high standard" was chosen as the origin: (3.7,3.7).

Alloway's labeling of the four quadrants is also used here. The quadrant where importance and satisfaction are both rated high is the "SUCCESS" quadrant, while the quadrant where both importance and are rated low is the "OK" quadrant. The WASTE quadrant includes those items which were rated as relatively unimportant but which were also rated high with respect to satisfaction. Finally, the quadrant where importance is high but satisfaction is low is, according to Alloway, "the real 'killer.'" "

Before interpreting Figure 6.6, it should be noted that the first 13 aspects of UDA (A to M) listed in Table 6.27 are strongly DP-influenced, whereas the last five (N to R) are not. The relevance of this distinction is explained below.

SUCCESS

The SUCCESS quadrant includes four aspects of UDA. Of these four aspects, only one, namely Ease of Use of the Tools (A) is DP-related. The three other aspects (O:Timeliness; P:Control over Application; R:Usefulness of Information) are not under DP's control or influence.

KILLER

On the other hand, all six UDA aspects contained in the KILLER quadrant are DP-influenced. That is, all the aspects which are very important to users and with which users are not very satisfied are DP-influenced. However, some caution is required in order to interpret this result. Individuals may have a tendency to systematically rating lower those aspects which are controlled by "outsiders" such as DP, while rating higher aspects which they themselves control.

Nevertheless, keeping in mind the above warning, Figure 6.6 suggests that there are six aspects which are important to users but with which users are less satisfied. These aspects are:

- K. Availability of consultants
- L. Computer response time
- M. Availability of terminals
- C. Quality of training with the use of the tools
- B. Awareness of the programming tools available
- H. Consultants knowledge of programming tools

As discussed in Chapter V, insuring an appropriate response time to users is an issue in several of the firms studied. Often, users have to "compete" with production systems for response time. People responsible for supporting users mentioned that a negotiation had to take place between themselves and Operations people in order to arrive at an acceptable response time. Availability of those who provide assistance may be a problem in some instances, since typically a small number of consultants support a proportionally large number of users. Moreover, in periods of growth of an IC, consultants are often busy

training new users, hence are not immediately available for providing debugging assistance. Since, typically, from the point of view of the person who requires assistance, a bug needs to be fixed right away, dissatisfaction may arise from waiting.

The issue of knowledge of the programming tools by consultants was raised in some of the firms studied. While some users spend a considerable amount of time using the tools and become quite sophisticated, consultants do not "practice" as much, since they are busy teaching fundamentals, preparing course material, helping to debug programs, and so on. As put by a DP manager: "It is hard to ask them [the consultants] to be as knowledgeable about a product as some users are. You cannot ask somebody who has stopped writing code to know everything about the product."

Figure 6.6 further suggests that improvements are required with respect to the awareness of the programming tools and the quality of training users receive. The former might require from DP to be more heavily involved in making presentations of UDA tools made available to the user community.

Quality of training is an issue which was raised during the interview phase of the study. The main issue with respect to training was found to be the relatively short period of time devoted to training with a given tool, together with the large amount of material covered during the training period. Users indicated that as a consequence, they did not have time to assimilate what they were

taught.

The sixth aspect of UDA which is part of the KILLER quadrant is Availability of terminals. This aspect was previously classified as being DP-influenced. However, in some instances the availability of terminals is not dependent only on DP. For instance, during the interviews some heavy users indicated that the fact that they had to share a terminal with other users in their department was sometimes a problem. In most cases, user departments can have their own terminals, provided that they cost-justify them and assume their costs. Consequently, if terminals are not as easily available as users may wish, the issue is more related to their own department than to DP.

WASTE

According to Alloway, the WASTE quadrant "should receive declining management attention. Any increase in efforts to improve performance here are a relative 'waste' of resources" (Alloway, 1980a, p.16). With respect to the present study, the two aspects which fall in this quadrant are not DP-influenced. Those aspects are N, Not having to communicate needs to a DP professional and Q, Cost savings due to the fact that the user and not DP develop applications. In the present case, it might be more appropriate to label the quadrant BY-PRODUCT OF UDA rather than WASTE.

OK

Finally, the OK quadrant is constituted by those aspects which are relatively less important to users and with which users are relatively less satisfied. There are six aspects in the OK quadrant here:

- I. Consultants' understanding of the applications
- E. Assistance during the design of the applications
- D. Assistance in the choice of which programming tool to use
- F. Debugging assistance
- J. People orientation of consultants

It should be noticed that all these aspects are related to the assistance provided to users. The fact that these aspects are rated relatively low in importance suggests that we have to deal with respondents who are fairly independent from DP. This independence is consistent with the fact that, as mentioned in Section 6.2.2, 75% of the respondents had at least done some programming or had very good knowledge of computers before developing applications.

In summary, the above discussed respondents' evaluation of 18 UDA aspects which were grouped into four categories. The SUCCESS category consists of those aspects for which both importance and satisfaction were rated the highest. The KILLER category includes aspects that are very important but with which users are less satisfied. The OK category encompasses aspects which are less important to users and with which they are less satisfied. Finally, the WASTE category is made up of aspects which are less important and with which users are

very satisfied.

Of the four aspects included in the SUCCESS category, only Ease of use of the programming tools is DP-influenced. The other three aspects pertain to the independence from DP due to UDA. The KILLER category contains six aspects, all of which are DP-influenced, and pertain to both the support provided by DP (such as Consultants' knowledge of the programming tools, User awareness of the tools) and to the environment setup. The OK category is also related to aspects of UDA support, mostly with respect to assistance provided during the development of applications. Finally, the WASTE category was re-named BY-PRODUCT OF UDA, since, the two aspects it includes being non DP-influenced, DP does not waste resources supporting them.

6.3 TESTING THE MODEL

Section 6.3.1 presents the results of bivariate data analysis performed in order to individually test the propositions embodied in the research model. Section 6.3.2 discusses the results of regression analysis performed in order to summarize the impact of several variables on user satisfaction.

6.3.1 Testing the Propositions

For the reader's convenience, the propositions which were tested are listed below.

Proposition 1: The higher the degree of user satisfaction with independence from DP, the higher the degree of user overall satisfaction with UDA.

Proposition 2: The higher the degree of user satisfaction with environment setup, the higher the degree of user overall satisfaction with UDA.

Proposition 3: The more user friendly a programming tool is perceived by the user, the higher the degree of user overall satisfaction with UDA.

Proposition 4: The more positive the user attitude toward UDA, the higher the degree of user satisfaction.

Proposition 5: The computer background of the user has a moderating effect on the user perception of user friendliness of programming tools.

Proposition 6: The computer background of user has a moderating effect on user attitude toward UDA.

Proposition 7: The higher the degree of user satisfaction with support, the higher the degree of user overall satisfaction with UDA.

Proposition 8: The better the fit between the Pull exerted by users and the Push exerted by DP, the higher the degree of overall satisfaction with UDA.

Proposition 9: The degree of Push exerted by DP will be higher in the cases of firms with a higher degree of DP readiness for change.

Proposition 10: The perceived degree of Push is positively correlated with the user satisfaction with support.

Composite scales were used to measure the main constructs of the model, namely, degree of push, degree of pull, user attitude, user friendliness, satisfaction with support, satisfaction with environment setup, and satisfaction with independence from DP. The scales and the tests for the reliability of the scales are discussed in detail in Appendix I. The reliability coefficients were found to be sufficient to use the composite scales in the remainder of the analyses. The

propositions were tested using either the Spearman rank correlation coefficient or the Mann-Whitney test. Table 6.28 formally states the propositions, the null hypothesis which was tested for each proposition, the test performed, the test result and significance, and the conclusion of the test.

Proposition 1. The first proposition suggests that the degree of user satisfaction with independence from DP due to UDA varies together with overall user satisfaction with UDA (Section IV: Q6) [9]. Thus the null hypothesis, H_0 , is that no relationship exists between user satisfaction with the independence from DP due to UDA and user overall satisfaction with UDA. The Spearman correlation coefficient between the two variables is .4047, at a level of significance of .001. This result allows for rejecting the null hypothesis in favor of the research hypothesis, H_1 (the level of significance being more stringent than .05) [10].

Proposition 2. The second proposition of the model implies that the degree of user satisfaction with the UDA environment setup varies with the degree of user satisfaction. As in the case of Proposition 1, the Spearman correlation coefficient was computed between the variables; the coefficient obtained was .2347, at the .001 significance level. While the degree of association between the variables is not as high as for Proposition 1, the relationship is statistically significant. Thus, the results support the proposition.

TABLE 6.28

SUMMARY OF RESULTS OF PROPOSITION TESTING

- PROPOSITION 1 : The higher the degree of user satisfaction with independence from DP, the higher the degree of user overall satisfaction with UDA.
- NULL HYPOTHESIS: User satisfaction independence from DP and overall user satisfaction are unrelated in the population.
- STATISTIC : Spearman rank correlation coefficient between user satisfaction with independence from DP, and user overall satisfaction with UDA.
- RESULT : .4047
- SIGNIFICANCE : .001 (one-tailed probability) N=265
- CONCLUSION : Reject H0 in favor of the Proposition.
- PROPOSITION 2 : The higher the degree of user satisfaction with environment setup, the higher the degree of overall user satisfaction with UDA.
- NULL HYPOTHESIS: User satisfaction with environment setup and overall user satisfaction with UDA are unrelated in the population.
- STATISTIC : Spearman rank correlation coefficient between user satisfaction with environment setup and user overall satisfaction with UDA.
- RESULT : .2347
- SIGNIFICANCE : .001 (one-tailed probability) N=266
- CONCLUSION : Reject H0 in favor of the Proposition.
- PROPOSITION 3 : The more user friendly a programming tool is perceived by the user, the higher the degree of overall user satisfaction with UDA.
- NULL HYPOTHESIS: User friendliness of programming tools and overall user satisfaction with UDA are unrelated in the population.
- STATISTIC : Spearman rank correlation coefficient between user friendliness and overall satisfaction with UDA.
- RESULT : .2499
- SIGNIFICANCE : .001 (one-tailed probability) N=264
- CONCLUSION : Reject H0 in favor of the Proposition.
- PROPOSITION 4 : The more positive the user attitude toward UDA, the higher the degree of user overall satisfaction.
- NULL HYPOTHESIS: User attitude toward UDA and overall user satisfaction with UDA are unrelated in the population.
- STATISTIC : Spearman rank correlation coefficient between attitude and overall satisfaction with UDA.
- RESULT : .2261
- SIGNIFICANCE : .001 (one-tailed probability) N=265
- CONCLUSION : Reject H0 in favor of the Proposition.
- PROPOSITION 5 : The computer background of the user has a moderating effect on the user perception of user friendliness of programming tools.
- NULL HYPOTHESIS: Computer background of user and user perception of user friendliness of a tool are unrelated in the population.
- STATISTIC : Spearman rank correlation coefficient between computer background and user friendliness
- RESULT : .3192
- SIGNIFICANCE : .001 (one-tailed probability) N=267
- CONCLUSION : Reject H0 in favor of the Proposition.

TABLE 6.28 (Continued)

PROPOSITION 6 : The computer background of a user has a moderating effect on user attitude on UDA.
 NULL HYPOTHESIS: User computer background and user attitude toward UDA are unrelated in the population.
 STATISTIC : Spearman rank correlation coefficient between computer background and attitude.
 RESULT : .2377
 SIGNIFICANCE : .001 (one-tailed probability) N=268
 CONCLUSION : Reject H0 in favor of the Proposition.

PROPOSITION 7 : The higher the degree user satisfaction with quality of support, the higher the degree of overall satisfaction with UDA.
 NULL HYPOTHESIS: User satisfaction with support and user satisfaction with UDA are unrelated in the population.
 STATISTIC : Spearman rank correlation coefficient between satisfaction with support and satisfaction with UDA.
 RESULTS : Satisfaction weighted by importance : .2798 SIGNIFICANCE : .001 (One-tailed probability) N=257
 Satisfaction not weighted : .4015 SIGNIFICANCE : .001 (one-tailed probability) N=258
 CONCLUSION : Reject H0 in favor of the Proposition.

PROPOSITION 8 : The better the fit (i.e. the smaller the value of the variable fit) between the Pull exerted by users and the Push exerted by DP, the higher the degree of user satisfaction with UDA.
 NULL HYPOTHESIS: The fit between Push and Pull and user satisfaction with UDA are unrelated in the population.
 STATISTIC : Spearman rank correlation coefficient Fit and overall satisfaction with UDA.
 RESULT : +.2431
 SIGNIFICANCE : .001 (one-tailed probability) N=195
 CONCLUSION : The results do not allow for rejecting H0.

PROPOSITION 9 : The degree of Push exerted by DP will be higher in the cases of firms with a higher degree of DP readiness for change.
 NULL HYPOTHESIS: The degree of Push exerted by DP is equal in the two types of firms.
 STATISTIC : Mann-Whitney between DP readiness for change and Push.
 RESULT : Group : Low Readiness High Readiness
 Mean Rank : 109.62 139.72
 N : 129 118
 SIGNIFICANCE : .001 (one-tailed probability)
 CONCLUSION : Reject H0 in favor of the Proposition.

PROPOSITION 10 : The perceived degree of Push is correlated with the perceived quality of support provided to users.
 NULL HYPOTHESIS: Perception of the degree of Push and satisfaction with support are unrelated in the population.
 STATISTIC : Spearman rank correlation coefficient between degree of Push and satisfaction with support.
 RESULT : Satisfaction with support weighted by importance : .3385
 SIGNIFICANCE : .001 (one-tailed probability) N=240
 Satisfaction with support not weighted by importance : .3738
 SIGNIFICANCE : .001 (one-tailed probability) N=241
 CONCLUSION : Reject H0 in favor of the Proposition.

Proposition 3. The third proposition suggests a positive relationship between the perception of user friendliness of the programming tools users use (Section IV: Q1B.) and the overall satisfaction with UDA. The Spearman rank correlation coefficient between user friendliness and overall satisfaction is .2499, with a significance of .001. This result supports the proposition.

Proposition 4. This proposition suggests that user attitude toward UDA and satisfaction with UDA vary together; that is, the more positive the user attitude toward UDA, the higher the degree of user satisfaction. The attitude scale is composed of six items which pertain to the user confidence in his ability to develop applications, the degree to which he likes or dislikes developing applications, and his attitude toward the role played by DP in the context of UDA (see Appendix I). A high score on the attitude scale would suggest that the respondent is very independent from DP, and that he would prefer to stay so. As indicated by Table 6.28, the Spearman correlation coefficient between user attitude and overall satisfaction with UDA (.2261) is significant at the .001 level, and thus supports the proposition.

Proposition 5. Proposition 5 states that the computer background of a user will have an influence on the user perception of user friendliness of programming tools. The Spearman correlation coefficient between the variables Computer Background and User Friendliness is .3192 (Significance .001). This indicates that the

better the user background with computers, the more user friendly the tools are rated. This result supports the proposition.

Proposition 6. The research model suggests that a user's computer background will have a moderating effect on that user's attitude toward UDA. The Spearman correlation coefficient between the two variables indicates that the stronger the user's computer background, the more positive his attitude toward UDA (Spearman .2377, Significance .001), and thus supports the proposition.

Proposition 7. The research model suggests that the quality of support provided to users has an impact on user satisfaction with UDA. In the present study, no objective measure was made of the quality of the support provided to users in each firm. Users were asked to evaluate the quality of support they received by indicating how satisfied or dissatisfied they were with nine aspects of UDA related to support (see Appendix I for a list). Consequently, the proposition tested pertains to the quality of support as perceived by users.

Table 6.28 indicates that there exists a positive relationship between the perceived quality of support and user satisfaction. It should be noticed that the Spearman coefficient of correlation was computed for two different support scales. The first scale is Satisfaction with Support weighted by Importance of Support for the user. Weighting was done by multiplying the importance score of one item by the satisfaction score of the same item [1]. The weighted

satisfaction scores were then added to obtain the final satisfaction score. In this case, the Spearman correlation coefficient between satisfaction with support and overall satisfaction is .2798, with a significance level of .001. on the other hand; the Spearman correlation coefficient between satisfaction with support, when it is not weighted by importance, and overall satisfaction is .4015 at the same level of significance.

While it was thought that weighting satisfaction with importance would give a more accurate image of satisfaction, the above results suggest that either the mode of weighting or weighting itself, do not provide as good a fit as using satisfaction unweighted.

Proposition 8. Proposition 8 pertains to the fit between the efforts made by DP toward UDA (Push), and the pressures exerted by users (Pull). As discussed in Appendix I, the fit between Push and Pull is measured by the absolute value of the difference between Push and Pull (ABS PUSH - PULL). A good fit exists when there is an equal amount of Push and Pull, that is, when Push - Pull equals 0. The proposition suggests that Fit and overall satisfaction will vary in opposite directions; that is, the lower the value taken by the variable Fit, the higher the satisfaction.

The Spearman correlation coefficient between Fit, as defined, and Overall Satisfaction does not confirm Proposition 8. As indicated by Table 6.28, the results suggest that the larger the difference between Push and Pull, the higher the satisfaction with UDA. Moreover, the

Spearman correlation coefficient between Push and satisfaction (.2955, Significance .001) and between Pull and satisfaction (-.0898; Significance .094) strongly suggest that Push is more closely related to satisfaction than Pull. That is, according to these results, a balance between the amount of pressure made by users and the amount of efforts toward UDA made by DP is not what counts with respect to satisfaction. It appears that what is most important is the amount of effort made by DP to facilitate UDA. However, the scale Pull included only two items, so such a claim is made with caution here.

Proposition 9. Proposition 9 suggests that the more DP management is ready for change, the higher the degree of Push will be. While no formal measurement of DP readiness for change was performed during the interview phase of the study, the 10 firms were subjectively classified into two categories. Those categories are "More Ready" and "Less Ready." The most important basis for classification was the interviews with DP managers. As the category names indicate, DP management in one company was classified after comparison with DP management in other companies. Such a comparison was based on investigator assimilation of factors such as the DP manager's perception of the pressures from both the user community and DP itself, his perception of the seriousness of the consequences of not going toward UDA, and his general attitude toward UDA.

The Mann-Whitney test indicates that there is a statistically significant difference between the two groups, with regard to the degree of Push, as perceived by users (Significance: .001). This result

supports Proposition 9.

Proposition 10. The perceived degree of Push is correlated with the perceived quality of support provided to users. As discussed previously, the degree of Push, as measured by the questionnaire, pertains to the efforts made by DP to encourage or facilitate UDA. As with Proposition 7, two Spearman correlation coefficients were computed, between Push and satisfaction with support weighted by importance (.3385) and with satisfaction with support, not weighted by importance (.3738). The latter is larger than the former, as was the case for Proposition 7. However, the difference is not as large. The results of the test support Proposition 10.

To sum up, while in absolute value, the coefficients of correlation obtained may appear to be low, they compare favorably to results obtained in similar types of research. Lucas (1978) summarizes the findings of nine empirical studies he conducted in order to test a model of MIS implementation. He reports correlation coefficients varying from .20 to .53. Those results are in the same range as the ones obtained in the present study. Moreover, the results obtained here suggest that while the degrees of association between the variables are not high, they are significant at a level as stringent as .001.

6.3.2 Regression Analysis

This section presents the results of regression analysis which was performed on the variables of the model. The primary reason for which regression analysis was performed was to summarize the impact of the model's independent variables on user overall satisfaction with

TABLE 6.29

REGRESSION MODEL

DEPENDENT VARIABLE.. OVERALL SATISFACTION WITH UDA

INDEPENDENT VARIABLES: SATISFACTION WITH SUPPORT
USER FRIENDLINESS OF TOOLS
USER ATTITUDE TOWARD UDA
DP READINESS FOR CHANGE
USER SATISFACTION WITH INDEPENDENCE FROM DP

MULTIPLE R 0.61119
R SQUARE 0.37355
STANDARD ERROR 0.61970

ANALYSIS OF VARIANCE	DF	SUM OF SQUARES	MEAN SQUARE	F
REGRESSION	5	55.189	11.038	28.74
RESIDUAL	241	92.552	0.384	

----- VARIABLES IN THE EQUATION -----

VARIABLE	B	BETA	STD ERROR B.	F
SUSAT	0.4657588	0.38126	0.06517	51.083
FRI1	0.2053511	0.14034	0.08024	6.549
ATT	0.2748380	0.18545	0.08530	10.381
READY	0.1920729	0.12417	0.07996	5.769
INSAT	0.3584853	0.25934	0.07544	22.583
(CONSTANT)	-0.9184218			

UDA, and to identify which variables were the most important in explaining the variance of user satisfaction. Table 6.29 outlines the results of the regression model which provided the best fit between the independent variables and satisfaction with UDA. Five variables contribute to explaining 37% of the variance of user overall satisfaction with UDA. Those five variables are: (1) user friendliness of the programming tools, (2) user attitude toward UDA, (3) user satisfaction with support, (4) user satisfaction with independence from DP, and (5) DP readiness for change.

The results obtained, then provide support for the User Satisfaction Sub-Model, at levels comparable to other similar research in this and related fields. However, they also suggest that more research is required to improve the model, that is, to increase the amount of explained variance of user satisfaction.

6.4 SUMMING UP THE SURVEY PHASE

The objective of the survey phase was to obtain additional information on UDA from a large number of users, and to test the relationships implied by the User Satisfaction Sub-Model. The results presented in Chapter VI are based on the responses of 272 users.

Section 6.1 provided elements of answer to questions such as: What type of individuals develop applications? What types of applications are developed and run? What are from a user point of view, the advantages and disadvantages of UDA? How is user

productivity affected by the applications developed? How do users evaluate their experience with UDA?

Section 6.2 presented the results of data analysis performed so as to test the relationships implied by the model. The results of bivariate analyses provided support for nine of the 10 propositions of the model. The results of regression analysis indicate that five variables contribute to explain 37% of the variance of user overall satisfaction with UDA. While providing support for the model, this result indicates that further research is required in order to improve the model so as to increase the amount of explained variance of user satisfaction.

The next chapter summarizes the research results, and discusses some implications for MIS management and for MIS research.

FOOTNOTES

[1] The appropriate Section and question number will be indicated between parentheses.

[2] See Appendix G, Company E, for more details.

[3] The category Analyst actually includes three sub-categories: Financial Analysts (4.5%), Analysts (20.0%), and Pseudo-Analysts (4.5%). The latter are those respondents whose formal title is Analyst but who have a DP professional-like response profile.

[4] All data analyses were performed using SPSS (Statistical Package for the Social Sciences), Release 8.1.

[5] While Scheffe's a posteriori test is available on SPSS with the subprogram ONEWAY, no such tests are available for the Chi-square and for Kruskal-Wallis tests. However, a posteriori tests were performed by repeating either the Chi-square or the Kruskal-Wallis by comparing only the groups between which appeared to be differences. The significance of these a posteriori tests was adjusted by the following formula:

S(N): LEVEL OF SIGNIFICANCE OF THE Nth TEST

$$S = 1 - (1 - S(1))(1 - S(2)) \dots (1 - S(N))$$

For instance, if two Chi-Square tests are performed on the same data, and if each test is significant at the .05 level, the overall significance is:

$$S = 1 - (1 - .05)(1 - .05)$$

$$S = 1 - .9025 = .0975$$

[6] As one might expect, there is a relationship between the number of months of experience with UDA and the number of applications that have been developed. The Pearson coefficient of correlation is .39.

[7] The number of missing cases may appear surprisingly high for the questions reported in Table 6.26. However, in each case, the large number of missing cases is due to the fact that the respondent does not develop the type of applications to which the question pertains.

[8] In order to present the results in the fashion proposed by Alloway (1980a), the variables pertaining to satisfaction and importance have been recoded, by inverting the scales. Consequently, on a 5 point scale, 5 indicates very important or very satisfied.

[9] The 7 point satisfaction scale was transformed into a 5 point scale in the following way: (1=5) (2,3=4) (4=3) (5,6=2) (7=1). The inversion of the scale was done to maintain consistency with the inversion of the importance and satisfaction scales discussed above. A better fit was obtained by collapsing (2,3) and (5,6).

[10] In the remainder of the presentation of the bivariate analysis results, the reader should interpret "the results support the proposition" as actually being "H0 is rejected in favor of H1." In this study, the level of significance for rejecting H0 was .05.

[11] Other research dealing with satisfaction and importance have used the same weighting approach (see Alloway, 1976, Thirkell, 1980).

CHAPTER VII

TAKING STOCK OF THE STUDY

Taking stock of: think carefully about, as to take stock of a person or situation (i.e. judge or form an estimate of him or it):

The Advanced Learner's Dictionary of Current English, Oxford University Press, 1958, p.1262.

The purpose of this chapter is to take stock of the present study. In order to do so, the research process that took place is first summarized, from the statement of the study's purpose through the presentation of the results of the empirical investigation. Second, the main implications of the study for the management of UDA are outlined. Finally, the study's implications for future research are discussed.

7.1 SUMMING UP THE RESEARCH PROCESS

The purpose of the present study was to develop and provide empirical evidence for a model of the factors of success of UDA. First, the MIS literature was reviewed, in order to "map the terrain" and establish the premises on which a model could be built. Second, the initial research model was developed, on the basis of the literature reviewed. In a third step, a field investigation was conducted in order to better understand UDA and the various organizational contexts in which it takes place, as well as to examine the initial research model in the light of the "real world." A major outcome of the field investigation was a revised version of the

research model. Finally, a survey was conducted. The survey questionnaire was designed in order to obtain additional descriptive information on UDA from a large number of users, and to gather data so as to test some of the relationships implied by the revised version of the model.

In summing up the research process, the following paragraphs briefly review the pre-investigation steps of the study and summarize the findings of the investigation phase.

7.1.1 Reviewing the Pre-Investigation Steps

As stated in Chapter I, UDA is a relatively new phenomenon, and up to now very little research has been done on it. Yet, according to some authors (McLean, 1979; McCracken, 1980; Martin, 1982), application development by end users has the potential of solving some of the "software development problem," and is likely to spread rapidly.

First, by helping relieve DP application development and maintenance workload, UDA is one way of dealing with the ongoing issue of DP professional productivity. Second, by eliminating the role of intermediary played by the systems designer, UDA eliminates those communication problems which often lead to difficulties in the determination of user information requirements and in the implementation of applications.

However, because the phenomenon is new and literature on it is scarce, organizations considering the introduction of UDA are faced with more questions than answers with regard to what should be done to make UDA a successful endeavor. The purpose of this study was to help organizations address this phenomenon, by developing and providing empirical evidence for a model of the factors of success of UDA.

The first task to be undertaken was to review the literature in order to map the terrain and to establish the premises for an initial research model. The review of the literature first focused on the phenomena from which UDA originates. The literature directly pertaining to UDA was surveyed, so as to come up with a definition of UDA success and to identify factors of success. Other related areas were also studied, and their relevance discussed.

The initial research model was the result of an effort to integrate the literature on UDA with the literature of other relevant areas, and to suggest conjectures with regard to the relationships between the different elements of the literature, and success of UDA. The model was used as a basis for investigation and was refined during field work.

The initial model defined success of UDA from two perspectives: from a user point of view and from a DP point of view. As suggested by the UDA literature, success as seen by DP was defined as the degree of diffusion of UDA, the reduction in the backlogs, and the decrease in the amount of DP resources devoted to maintenance. Success from

the user vantage point was measured by user satisfaction with UDA and with the organizational context in which it takes place.

The model further identified seven factors of success for UDA. User friendliness of the programming tools available to users, the type of controls exerted by DP over UDA, and the quality of support provided by users are factors which were suggested by the UDA literature. DP readiness for change and user attitude toward UDA were suggested by the literature on organization change. Finally, two dimensions were found to be important but were not found in the literature reviewed. Those dimensions were the degree of maturity of DP and the strategy by which UDA is introduced in the organization. The model identified two types of strategies, namely Pull and Push. A Push strategy exists when DP initiates UDA, while a Pull strategy exists when users start developing applications without DP support, and sometimes in spite of it.

In choosing a research approach, the fact that other, yet unknown, factors might be important in the context of UDA had to be taken into account. That is, because UDA is a new phenomenon it was believed that this study should put as much emphasis on theory generating as on theory testing. In order to achieve these two objectives, the study was conducted in phases and took a multi-method approach.

Ten firms participated in the study. In each organization, qualitative research methods such as structured and unstructured

interviews, study of documents, observation, and comparative analysis were first used. The qualitative approach was chosen in order to obtain a holistic view of UDA and of its context in each firm studied, to remain flexible, and to allow for exploration. This phase had two major outcomes, namely, a description of UDA in each of the firms studied and a revised, improved version of the research model. A survey questionnaire was then sent to users who develop applications in each of the 10 participating companies. The purpose of this survey was to obtain additional descriptive information from a large number of users, and to test some of the relationships implied by the revised version of the research model.

7.1.2 Summary of the Research Findings

The investigation effort for this study took place in three phases: (1) the identification phase, (2) the interview phase, and (3) the survey phase.

The Identification Phase

The purpose of the identification phase was to locate organizations where UDA existed. In order to do so, a letter explaining the research project and defining UDA was sent to DP managers of the 150 largest Canadian companies, according to Canadian Business, July 1981. Along with the letter was a short questionnaire asking the DP managers if UDA existed in their organization, and if so, how many users were involved therein. From the 150 questionnaires sent, 108 responses were received, out of which 25 (23%) had 50 or

more users who developed applications. The 10 organizations which were to be part of the study were chosen from among those 25 firms.

The Interview Phase

The interview phase was undertaken in order to obtain empirical data on UDA. In turn, these data would help better understand the phenomenon as well as various organizational contexts in which it can take place. Moreover, the interview phase was found to be the most appropriate way of submitting the initial research model to the "acid test" of the real world. During the interview phase, two main questions were addressed: (1) Is there empirical evidence to support each of the factors of success included in the initial model? and (2) Does the experience the firms in the sample have had with UDA suggest the inclusion of other factors in the model?

The major products of the interview phase are descriptive information on UDA in the ten firms studied, and a revised version of the research model. This section briefly describes the background information about the organizations studied and summarizes a user taxonomy which was developed during this phase of the study. Chapter V and Appendix F provide a more detailed view of UDA in the 10 firms studied. The following paragraphs also discuss the revised version of the research model.

The firms studied were from various industrial sectors: financial institutions (2), insurance industry (1), manufacturing (3), utility (2), communications (1), and forest industry (1). The assets

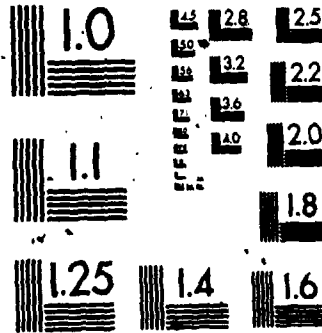
of the companies studied vary from \$900 million to more than \$50 billion. The firms employed between 2,500 and 40,000 people.

All the organizations studied have been using computers for commercial DP for more than 15 years. In all 10 firms some support was provided by DP to users who developed applications. In seven of the companies, responsibility for supporting users belonged to an Information Centre (IC) or to a formal support group whose responsibilities were similar to those of an IC. The other three organizations provided support through "product coordinators."

A total of 64 users were interviewed in the 10 organizations. Early in the interview phase patterns started to emerge, with regard to the reasons for which users develop applications, for whom they developed applications, and so on. At this stage it appeared relevant to develop a taxonomy of users, so as to better understand who the users are. The following user categories were delineated: (1) Micro DP Department (42.2% of users interviewed), (2) Staff Analyst (31.3%), and (3) Opportunity Seeker (26.6%).

Micro DP Department includes users whose main role in the context of UDA is to respond to requests for applications from other users. For most users in this category, developing applications for others is a formal duty. Those users were called Micro DP Department because their task is very similar, although on a smaller scale, to that of the main DP group. Moreover, they sometimes have to face problems similar to DP's, such as backlog of applications and numerous requests.

4



for maintenance.

The second category is Staff Analyst. In most cases, users in this category see UDA as a means to an end, not an end in itself. Typically, the formal responsibilities of Staff Analyst type of users are to perform studies, the results of which will be used by management to control, plan, and make decisions. The applications they develop generally are tools they use to perform their studies.

Finally, the third category, Opportunity Seekers, consists of individuals who have much expertise in their own functions, who often are at the managerial level (such as supervisor or budget director), and who generally belong to a staff department of the organization. While they have some similarities with Staff Analysts, they appear to be more independent, more creative, and more proactive, in the sense that they actively identify problems which could be solved by using the computer.

As a result of the perspective gained in the interview phase, the initial research model was modified. The most important modifications concerned changes in the definition of success of UDA from a DP perspective. While the initial research model defined success in terms of the decrease of backlogs and the decrease of the maintenance load, it was found during the field work that this is not the way DP managers actually perceive UDA success. For seven firms out of the 10, the primary measure of success is the "favorable cost benefits of UDA," which include favorable cost benefits of the applications

developed by users and/or reduction of use of outside timesharing services. For the remaining three organizations DP defines UDA success as "user satisfaction." Interviews with users confirmed the appropriateness of user satisfaction as a measure of success from a user perspective.

Analysis of the data gathered in the interview phase suggested the partitioning of the research model into two sub-models which, for the purpose of this study, were dealt with independently. These were the User Satisfaction Sub-Model and the Favorable Cost Benefits of UDA Sub-Model.

The User Satisfaction Sub-Model was a refinement of the initial research model. Based on the findings of the interview phase, the initial research model was revised as follows. One variable, the degree of maturity of DP, was abandoned. Another variable, the strategy by which UDA is introduced was refined and split into its components. Three new variables were added, namely, user satisfaction with environment setup, user satisfaction with independence from DP, and user computer background. The remaining variables of the User Satisfaction Sub-Model were part of the initial research model. They were: perception of user friendliness of the tools, user attitude toward UDA, DP readiness for change, and user satisfaction with support.

The Favorable Cost Benefits of UDA Sub-Model focuses on UDA evaluation, and stresses the importance of planning for UDA

evaluation. The model was derived from Keen's (1975) "ideal agenda for negotiation," in the context of Decision Support Systems evaluation. The Favorable Cost Benefits of UDA Sub-Model includes the following four critical steps: (1) Define UDA Success, (2) Develop Methods and Mechanisms for Evaluation, (3) Reach a Consensus with Top Management, and (4) Apply the Evaluation Methods and Implement the Evaluation Mechanisms. This sub-model was not investigated further in this study.

The Survey Phase

The survey phase was conducted in order to obtain additional descriptive information on UDA from a large number of users, and to test the relationships implied by the User Satisfaction Sub-Model. Chapter VI presents and discusses the results of this phase of the study.

The descriptive statistics obtained from the survey provided information on the users themselves, the type of applications they develop, how the applications are developed and run, the advantages and disadvantages of UDA, and how users evaluate their experience with UDA.

Respondents to the questionnaire were classified according to the three-way taxonomy developed during the interview phase. The proportion of respondents in each category was similar to the proportion noted during the interviews. The three categories of users were compared on 13 variables, five of which were used to classify

them. For eight variables out of 13, responses of the Micro DP Department users differed significantly from each of the other two groups. For two other variables, the Micro DP Department category was significantly different from at least one of the other two categories. On the other hand, there was a statistically significant difference between Opportunity Seekers and Staff/Analyst on three variables only: the length of experience with UDA, the number of applications developed, and the job prior to beginning application development. These results support the observation made as a result of the interview phase, that Staff Analyst and Opportunity Seekers have many affinities. Chapter VI further suggests that Opportunity Seeker might be a natural "next career step" for the Staff Analyst category.

Users were found to develop a broad variety of applications, from model building to simple extraction of data. Applications which provide organizational support were least common, while group support and individual support applications received roughly equal user development attention. Most applications users develop (66.8%) are run on a regular basis.

In general, users tend to develop applications alone rather than in teams. Respondents to the survey questionnaire generally use their company's central computer to develop and run their applications, and UDA activities involving the use of terminals are most often performed using a terminal users share with others in their departments. Fully 98 different programming tools were mentioned by respondents as being the tool most often used. However, 12 of the 98 tools account for 82%

of the respondents. Only one percent of the applications users develop are run by the user's superior. In contrast, nearly 41% of the applications developed by Opportunity Seekers are run by their subordinates (as opposed to 9% for Staff Analyst and 17% for Micro DP Departments).

The timeliness with which applications are developed was cited by 50.6% of the respondents as one major advantage of UDA. Other advantages are the ability for the user to use his own expertise in order to arrive to an application which does exactly what it has to do, the independence and increased control of users over the development process, flexibility, cost reduction, and the fact that there are no communication problems between user and developer.

Ten percent of the respondents indicated that UDA was without disadvantages, while one third identified their lack of expertise with programming and the use of computers as being a major disadvantage. The time required to develop applications, the lack of efficiency of user written code, security and validity issues, and DP related issues were other disadvantages mentioned by respondents.

Some results suggested that there were important user productivity improvements aspects to UDA. An "average productivity improvement factor" was calculated for three types of applications, namely, the one-shot application which required the least time to develop, the one-shot application which took the longest period of time to develop, and the average regularly run application. For the

purposes of the present study, productivity improvement factor is defined as "how much longer it used to take (or would have taken) to perform the same task without a given application." An average productivity improvement factor of 4.9 was found for the shortest one-shot application, and of 6.2 for the longest one-shot application. The average productivity factor for regularly run applications is 6.2.

Users were found to be generally satisfied with UDA. User satisfaction was measured, by combining 18 different aspects of UDA. On a 5 point scale, none of the average satisfaction scores is lower than 3.1. Aspects which were rated as being very important and with which users were very satisfied are: (1) ease of use of the programming tools, (2) timeliness of application development, (3) control over applications, and (4) usefulness of the information provided by the applications. On the other hand, aspects which users rated as being important but with which they were less satisfied are: (1) the availability of those who provide assistance, (2) response time, (3) availability of terminals, (4) quality of training with the programming tools, (5) awareness of the programming tools available, and (6) knowledge of the programming tools by those who provide assistance to users.

Data analysis was performed in order to test the relationships implied by the User Satisfaction Sub-Model. Bivariate analysis was first performed, using non-parametric statistics. Nine of the eleven propositions of the model are supported by the results. Regression analysis was also performed in order to summarize the relationships

between overall satisfaction with UDA and the independent variables. Five independent variables explain 37% of the variance of overall satisfaction with UDA. Those variables are: user friendliness of the tools, user attitude toward UDA, DP readiness for change, satisfaction with support, and satisfaction with independence.

7.2 MANAGERIAL IMPLICATIONS OF THE RESEARCH FINDINGS

The following discussion of the managerial implications of the study emphasizes the practicality of the findings. Drawing from the experience of the organizations studied, three aspects of UDA have been identified as critical factors for successfully managing UDA. Those critical factors are (1) the UDA evaluation process, (2) the importance of determining user requirements with respect to software tools, before introducing such tools, and (3) the importance of the support provided to users.

7.2.1 The UDA Evaluation Process

An important implication of the study findings is that, from a DP point of view, the introduction of UDA and of the associated tools and support should be managed with as much care as any other business endeavor. In most of the organizations studied, top management put emphasis on having DP provide a cost justification for any major increase in resources required to accommodate UDA. Furthermore, several DP departments were also responsible for demonstrating to top management that the use made by end users of the computer resources is

profitable for the organization.

Some DP departments had stated precise objectives for UDA, had reached an agreement with top management on these objectives, as well as implemented mechanisms to measure performance with respect to the UDA objectives. On the other hand, in other firms it was found that, either the criteria for evaluating UDA success were not consistent with the reasons for which it had been introduced, or that the methods and mechanisms required for evaluating UDA were not applied.

What matters here is not the UDA objectives themselves; they may be as diverse as decreasing outside timesharing usage, increasing user productivity, improving user satisfaction, and so on. What is critical for DP is, first to clearly define UDA success as well as methods and mechanisms which will be used for UDA evaluation. Second, DP ought to reach an agreement with top management with regard to the definition of success and with the methods and mechanisms of evaluation. This agreement should be arrived at before UDA is introduced. Finally, the methods and mechanisms of evaluation should be implemented along with the UDA tools and support.

7.2.2 Determining User Requirements in Terms of UDA Tools

During the study it was found that, typically, DP puts little effort into determining which programming tools are best suited to user needs. Apart from one or two exceptions, DP acquired the tools which were available from their hardware vendor. Without arguing

about the quality of those tools, it might be appropriate for DP to do some "tool searching" as well as some "user needs surveying" before deciding on a tool or a set of tools. The importance of doing so was suggested by comments made by some users in the organizations studied. Generally, users felt that DP was not aware of what tools they really needed: "They [DP] know that we want to do some of the job ourselves. They come with products, without even asking us what we need."

Some problems may derive from this oversight from DP. In some cases, DP was concerned by the fact that some users were developing applications which were not appropriate to the tools they used. For instance, users might use ADRS-II to develop fairly large transaction processing applications. Doing so created difficulties. The user often reached a point where his program became so complex (given his DP expertise and the tool's features) that it was nearly impossible for him to deal with it. If he decided to abandon the development of the application, the user felt that he had wasted time, and sometimes put the blame on DP. Very often, the DP support staff would have spent several hours helping the user debug the application. If the user decided to forgo the development of such an application, much time had been wasted for the support staff as well.

While the above is an extreme case, it illustrates the risks related to a mismatch between user needs and UDA tools. Interestingly enough, while the comment that users are developing applications which are not appropriate to the tools they are using was made by a few of the DP people interviewed, none stated that DP does not provide users

with the appropriate tools they need to develop their applications.

7.2.3 Importance of Support Provided to Users

Both the interview phase and the survey phase of this study suggest that the quality of support provided to users is a critical factor for successfully managing UDA. Since the quality of support is directly dependent on the individuals who provide it, UDA puts very high demands on the consultants. Typically, when a support group such as an Information Centre (IC) is implemented, each consultant is expected to be proficient with at least one tool, to have the ability to teach; to know well the company's information systems, to be available, to be people oriented, and so on. Since it might be relatively difficult to staff an IC with individuals who combine all of the above characteristics, some of the qualities which were found to be the most important are discussed below.

In the organizations studied, a common requirements for IC staff is the ability to relate easily with people. As put by a DP manager, "the IC has no use for misanthropes." Good knowledge of at least one of the programming tools, and "backup" knowledge of a second one is another requirement common to the firms studied.

From a user point of view, the survey indicated that the consultants' knowledge of the programming tools is the most important aspect of UDA, Consultants' availability and the quality of the training provided were also found to be very important to users.

Typically, it is the responsibility of the IC staff to make the availability of programming tools known to the user community. To the users who completed the survey questionnaire, awareness of the programming tools available is very important. However this, and the assistance in the choice of the proper programming tool to use, were two aspects of UDA with which users were less satisfied.

The following user comment illustrates the frustrations users may experience when they first enter the world of UDA.

I took the GIS course, of approximately four days, and concluded it was of little use in our work. Also, I could not get the time to go to the Information Centre and flounder through hours of errors, finding out what the machine accepts and rejects, the recommended system of learning.

CMS came next. I took the half day course, but could not get the time to practice on a terminal using the 355 page self help manual.

I discussed the problem we have with the personnel in the IC, and PLANCODE was recommended. I read all the manuals and then the person in the IC knowing PLANCODE was transferred and it was no longer supported. Having seen the hours spent by the other learners in debugging programmes, I did not continue it.

SAS was recently recommended and I read all the manuals for it, but have not the required hours to experiment on the terminals....We recently had our junior member start with GIS and SAS, but it will be a long while before I can get any programme written for my work.

7.3 IMPLICATIONS FOR MIS RESEARCH

Several new issues for further research arose during the study. Below are discussed four research questions, the investigation of which would be important contributions to the MIS field. Those research questions pertain to (1) the economics of UDA, (2) the role of Data Administration in the context of UDA, (3) the broadening of the sampling frame, and (4) the development of objective measures for independence brought by UDA, for quality of support, and for quality of the environment setup.

7.3.1 The Economics of UDA

As previously discussed, the economics of UDA is of major importance to both DP and top management of the firms studied. In seven of the 10 organizations, the definition of UDA success, from a DP point of view, was related to the monetary benefits of UDA. However, the definition of monetary benefits was restricted to the favorable cost benefits of the applications developed by users and/or to the decrease of outside timesharing usage. While the UDA literature suggests that UDA has the potential of decreasing DP backlog and maintenance load, the organizations studied were measuring none of them.

An investigation into the economics of UDA would have two major tasks. The first one would be to investigate methods and mechanisms for helping DP in their task of demonstrating to top management that

the use which end users make of the computer is indeed profitable to the entire organization. While in most firms DP had the "fardeau de la preuve" with respect to favorable cost benefits of UDA, only one of the organizations studied had formal and operational mechanisms wherein users would provide DP with periodical cost benefits analysis of the applications they had developed.

While having affinities with research conducted on the evaluation of information systems, such an investigation would also have to incorporate other areas (e.g. chargeout schemes). Moreover, attention should be devoted to the study of user productivity improvements. While the present study found that UDA can indeed produce improvements in user productivity, it did not cover aspects such as the cost of support staff, as well as the cost of computer resources.

A second avenue of research pertaining to the economics of UDA is the investigation of the impact of UDA on DP. While the present study abandoned the decrease of DP application backlog and maintenance load as measures of UDA success, this is not to say that UDA does not have any impact on them. A study of the economic impact of UDA on DP ought to include those two aspects. Moreover, Rosenberger (1981b) identifies several potential DP productivity improvement areas. Those areas could serve as a starting point for focusing research on UDA impacts on DP.

7.3.2 Data Administration and UDA

The literature on UDA stresses the importance of Data Administration as a means of coordinating and facilitating user access to, and sharing of, data. As put by Martin, if the concept and the tools of Data Administration are not in place, UDA bears the risk of creating situations where "the same types of data field may be created many times by different groups. These fields will have different structures. Data that ought to be coordinated will not be. Data that ought to be passed from the factory floor...to other information systems will not be." (Martin, 1982, p.265)

In this study, Data Base Management Systems (DBMS) were in place in nine of the 10 companies. However, in most cases the technology only had been implemented, as opposed to the concept of Data Administration. While concerns existed with respect to data duplication, such concerns were not the most critical ones. However, since this research did not focus on the role of Data Administration in the context of UDA, further investigation is required into the relationships between the two concepts.

A comparative approach would allow the researcher to focus on the role played by Data Administration in UDA situations. Such an analysis would involve organizations wherein Data Administration exists, and organizations where it does not. UDA would be common to both groups. The experience of this study as well as the experience of other researchers (McCririck and Goldstein, 1980; Huff,

1981; Talbot, 1982) suggests that the term Data Administration is often used for the data base technology. Since the presence of data base technology is not sufficient for insuring data coordination and sharing, much care would be required to select the organizations to be part of each group studied.

While the UDA literature puts emphasis on the role played by Data Administration with respect to UDA, the following comment, from Joseph Ferreira, Vice-president of The Diebold Group, provides a new perspective from which to study the relationship Data Administration-UDA.

MIS departments can have a leading role in IRM [Information Resource Management], but it won't happen automatically. Unless MIS department heads take steps to free themselves of a lot of routine, time-consuming, low-payoff activities that clog their work load, they won't be able to involve themselves in new business tools that are a critical part of the IRM picture. The key to success will be offloading as much routine work as possible to the users and getting maximum user involvement in systems development. (Datamation, June 1981, p.75)

7.3.3 Broadening the Sampling Frame

The sample size of this study, in terms of the number of organizations, was relatively small (10 firms). The investigation of a larger number of organizations would allow for including firms where DP does not provide support to users. It was the intent of this study to examine firms wherein DP was facilitating UDA and firms where users developed applications without DP providing assistance (or even in spite of DP).

The actual sample of 10 firms included no organizations where the users who developed applications did so in spite of DP. While some DP groups provided somewhat less encouragement than others, they all provided users with some types of assistance.

After studying the first few companies during the interview phase, it appeared that even in an organization where UDA is officially supported by DP there may exist a disequilibrium between what DP is willing to give (Push) and what users ask for (Pull). The equilibrium between Push and Pull was termed the goodness as fit. The decision was then made to focus on those aspects within relatively similar contexts, that is, in firms where DP provides support to users.

Further investigations of UDA which focus on firms where users are not supported by DP in their application development activities would contribute much to an understanding of the evolution of UDA in organizations.

7.3.4 Development of Objective Measures

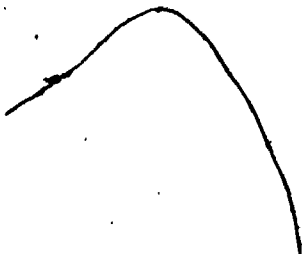
This study used "user satisfaction with..." as a way of measuring success for each of (1) degree of user independence from DP due to UDA, (2) quality of support provided to users, and (3) quality of the environment setup. All three variables were found to be positively correlated with overall user satisfaction with UDA. The first two variables, namely, user satisfaction with independence from

UDA and user satisfaction with support, are part of the regression equation which explains 37% of the variance of user overall satisfaction.

In order to pursue further the investigation of the User Satisfaction Sub-Model, it would be appropriate to develop more direct and objective measures for the variables mentioned above. Such measures would have to be composite scales, since each construct includes several dimensions. Quality of support provided to users, for instance, might involve the ratio user/consultants, the "turnaround time" to obtain a response to a request for assistance, the number of days devoted to training with a given tool, and so on.

Each construct would have to be investigated in detail so as to obtain an adequate measure. Ideally, such a measure should be as objective as possible, that is, should not involve any user judgement of how good or bad an aspect is.

This refining of the measurement of the above constructs might contribute to increase the amount of the variance of user overall satisfaction with UDA which is explained by the variables in the model. As such, the model itself would be improved.



7.4 CONCLUSION

This study investigated situations where non-Data Processing people in organizations develop computer-based applications either for themselves or for other end users. The purpose of the study was to develop and to provide empirical evidence for a model of the factors of success for UDA. An initial research model was developed, which integrated the UDA literature with literature from other relevant areas. This research model was used as a basis for conducting a field investigation, the outcome of which was a revised version of the research model. This revised version consisted of two sub-models, which, within the scope of the present study, were treated independently. The Favorable Cost Benefits of UDA Sub-Model stresses the importance, for DP, of the UDA evaluation process. The User Satisfaction Sub-Model defines UDA success from a user point of view. The survey phase, which followed the field investigation, was aimed at obtaining additional information on UDA, from a user point of view, and at testing the relationships implied by the User Satisfaction Sub-Model. While the results obtained provide support for nine of the 10 propositions included in the model, more research is needed to improve knowledge and understanding of UDA. Areas for future research have been identified, which include the broadening of the sampling frame so as to include firms where DP does not provide support to users, the investigation into the economics of UDA, and the study of the role of Data Administration in the context of UDA.

USER DEVELOPED COMPUTER-BASED APPLICATIONS:
A MODEL OF THE FACTORS OF SUCCESS

VOLUME II
APPENDICES

by

Suzanne Rivard

School of Business Administration

Submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy

Faculty of Graduate Studies
The University of Western Ontario
London, Ontario
December, 1982

© Suzanne Rivard 1982.

TABLE OF CONTENTS

APPENDICES

APPENDIX A. QUESTIONNAIRE AND LETTER: PHASE I.....	296
APPENDIX B. LETTER, EXECUTIVE SUMMARY, AND CHECK-LIST FOR DP INTERVIEWS.....	302
APPENDIX C. DP QUESTIONNAIRE.....	309
APPENDIX D. CHECK-LIST, FOR USER INTERVIEWS.....	332
APPENDIX E. SURVEY QUESTIONNAIRE.....	335
APPENDIX F. SCENARIOS OF UDA.....	352
APPENDIX G. PROGRAMMING TOOLS.....	462
APPENDIX H. REMINDER.....	468
APPENDIX I. COMPOSITE SCALES.....	470
BIBLIOGRAPHY.....	477
VITA.....	488

APPENDIX A

QUESTIONNAIRE AND LETTER: PHASE I



The University of Western Ontario

DOCTORAL PROGRAM RESEARCH
School of Business Administration
London, Canada
N6A 3K7

October 6, 1981

Dear

I am conducting a research project to investigate situations where Data Processing users develop their own administrative (as distinct from scientific or engineering) computer-based applications. The purpose of the research is to identify circumstances in which an organization should consider introducing this approach, the benefits it can bring as well as the issues which have to be faced.

Administrative application development by end users is a relatively new phenomenon. The first phase of this study is aimed at identifying organizations where this practice exists. To do this I would like to ask you to complete the attached short questionnaire. The second stage of the project will consist of gathering more information in organizations where end users develop administrative applications. Toward this end, the questionnaire asks you to indicate the name of a person I could contact in your organization.

The questionnaire should take only a few minutes to complete. I would be most grateful if you would complete it and return it as soon as your schedule permits. You may be assured of complete confidentiality. Information gathered from this questionnaire will be revealed in aggregate form only. No information on individual organizations or people will be divulged. A summary of the research findings will be sent to all respondents requesting it.

.../2

May I stress the importance of timing. The schedule set for my research project calls for the completion of this phase within a few weeks.

I appreciate your assistance in this study, which I am doing as a Ph.D. candidate at the School of Business Administration, The University of Western Ontario. This research project is my thesis and is a major requirement for earning my degree.

Sincerely,

Suzanne Rivard
Ph.D. Candidate
(519) 679-6055

Encl.

USER DEVELOPED COMPUTER-BASED APPLICATIONS QUESTIONNAIRE

The purpose of this questionnaire is to identify organizations where end users develop computer-based applications. The following definitions identify the type of users and the applications this study is concerned with, as well by what is meant by applications development.

END USERS: In the present study an end user is defined as being a member of a department other than Data Processing (or Information Systems), and for whom development of computer-based applications is not a full time activity or main function.

ADMINISTRATIVE COMPUTER-BASED APPLICATIONS: This study is concerned with business (or administrative) types of applications, as opposed to scientific or engineering types of applications.

APPLICATIONS DEVELOPMENT: For the purposes of the present study, two types of activities constitute applications development. These are:

1. Applications Maintenance

Making changes in applications which already exist and which have been developed by the organization's Information Systems department, or by users. These changes may be required because the user's environment, task, or other needs have changed.

2. Development of New Applications

Examples include:

- a. Writing a few (or several) lines of code, using a query language, to retrieve data from a database.
- b. Writing a program, using a general purpose programming language (such as APL, BASIC, COBOL) to perform simple or sophisticated calculations.
- c. Writing programs for, or interacting with, a statistical package (such as SPSS, SAS) to analyze data.
- d. Writing programs, using a report generator (such as EASYTRIEVE or RPGII), to print reports.
- e. Developing models, using a modeling language (such as IFPS, PLANCODE).

1. Given the above definitions, are there in your organization end users who develop computer-based applications?

— Yes

— No

If your answer is no, please go directly to question 4.

2. If yes, would you please estimate:

a) The number of end users who develop administrative computer-based applications. _____

b) The percentage of new applications which are developed by end users. _____

c) The percentage of the perfective maintenance which is done by end users. _____

3. Would you indicate a person I could contact in order to obtain more information about user developed applications in your organization.

Yourself _____

or Name: _____

Title: _____

Address: _____

Telephone: _____

4. Even if you answered No to question 1, I would appreciate it if you would indicate your name, title, and some information about your organization. Your assistance would help me to control the results of this phase of the study.

Name: _____
Title: _____
Organization: _____
Address: _____

Telephone: _____

Thank you very much for your assistance in this project. If you would like to receive a summary of the results of this study, please check the box below.

Please return the completed questionnaire to:

Suzanne Rivard
Doctoral Program Research
School of Business Administration
The University of Western Ontario
London, Ontario
N6A 3K7

Telephone: (519) 679-6055

APPENDIX B

LETTER, EXECUTIVE SUMMARY, AND
CHECK-LIST FOR DP INTERVIEWS



The University of Western Ontario

DOCTORAL PROGRAM RESEARCH
School of Business Administration
London, Canada
N6A 3K7

Dear Mr.:

Further to our recent telephone conversation I am sending you a summary of my proposed research project on application development by end users, a list of the questions I would like to discuss, as well as a copy of the questionnaire I will distribute to users who develop applications. This should provide additional information on the type of investigation I am undertaking.

All the information that I will gather during my research project will be held in strict confidence. No information that may be identified with individual organizations or people will be divulged without the explicit consent of the concerned organization and person.

If you should need further information, do not hesitate to contact me.

Sincerely,

Suzanne Rivard
Ph.D. Candidate
(519) 679-6055

RESEARCH ON USER DEVELOPED COMPUTER-BASED APPLICATIONS
EXECUTIVE SUMMARY

The following presents a plan for a thesis research on User Developed Computer-Based Applications (UDA). The purpose of the study is to acquire understanding of this phenomenon and to provide insights to assist Information Systems managers in with it.

Definitions

UDA. For the purposes of this study, UDA is said to take place when users develop (identify the information requirements, design, program, and implement) administrative applications or do perfective maintenance on such applications, either for themselves or for other end users.

Programs. The word program is used as a general term to describe a set of instructions which a computer will execute. These instructions may be written using general purpose programming languages, statistical packages, data base query languages, modeling languages, report generators, etc.

Programming Tools. The phrase programming tools will be used as a synonym for general purpose programming languages such as APL, BASIC, COBOL, FORTRAN, etc., data base query languages such as GIS, SEQUEL, NATURAL, etc., software packages such as GENMOD, IFPS, MAPPER, PLANCODE, SAS, SPSS, etc., report generators such as RPGII, EASYTRIEVE, and so on.

Computer-based applications. For the purposes of this questionnaire, the phrase application development activities includes tasks such as:

Development of New Computer-Based Applications

- a. Creating programs, using a query language (such as GIS, SEQUEL, NATURAL) to retrieve data from a data base.
- b. Creating programs, using a general purpose programming language (such as APL, BASIC, COBOL).
- c. Creating programs using a statistical package (such as SPSS, SAS) to analyze data.
- d. Creating programs, using a report generator (such as RPGII, EASYTRIEVE) to produce reports.
- e. Developing models, using a modeling language (such as IFPS, PLANCODE).
- f. Preparing graphics using a graphics package (such as SAS/GRAPH).

Applications Maintenance

Making changes in applications which already exist. These changes may be required because the user's environment, tasks, or needs have changed.

Questions addressed by the study

Because UDA is a relatively new phenomenon, little is known about the circumstances under which an organization should consider introducing it. The main questions the study addresses are:

- In which situations can UDA be beneficial to an organization, and conversely, in which situations can UDA be disruptive?
- When should an organization consider introducing the concept of UDA?
- What types of controls should be established by the Information Systems department over UDA?
- To what extent should the Information Systems department provide support to users when they develop applications?
- How should the integrity and security of corporate data be protected when users are allowed access to it directly?
- Which type of tools should be made available to users, and in what manner?
- Are there measurable benefits to the Information Systems department? To the users community? How might such benefits be measured?

Research Approach

Several organizations will be studied. In each organization, data will be gathered from two different sources: from the Information Systems department and from the user community. Interviews will take place with the manager of Information Systems, with the person responsible for user oriented services, where such services exist, and with representatives of the user community. Questionnaire will be distributed to users who will be asked to return them directly to the investigator, in order to ensure them of confidentiality.

All information provided by any source in the participating organizations will be kept confidential. No information that may be identified with individual organizations or people will be divulged without the explicit permission of the concerned organization and person. Once the study is completed, a research report will be provided to the participating organizations. This investigator will be available for discussing the research findings with the representatives of the Information Systems department of these organizations.

SUZANNE RIVARD
Ph.D. Candidate
School of Business Administration
The University of Western Ontario

LIST OF QUESTIONS

Information Systems Department

A. Information Systems Profile

How is IS organized?

How many people work in the department?

Does IS planning take place?

Does Data Administration exist?

Are users charged for systems development, use of computer resources?

B. Systems Development

Does IS make use of formal planning tools (such as PERT, CPM) for project development?

Are there written standards for systems development, programming?

What is the usual role of end users in the context of systems development?

C. UDA in the Firm

-Dispersion of UDA.

What is the number of users who do application development?

What is the potential number of users who could do application development?

What is the proportion of the firm's computer resources used by the users who develop applications?

What is the proportion of these users per department?

What has been the growth rate of the 'users-developers' since UDA started?

-Support to users.

Is there a support group for users who develop applications? If yes, when was the group formed? What is the budget of this group? Where is the group situated in the IS organization chart? What is the mandate of the group? What type of assistance does the group provide to users? What are the responsibilities of the staff?

-Controls.

Are there any restrictions on the type of applications users can develop? Are users charged for assistance they receive, CPU, terminal connect time (other)? Do users have to cost justify the applications they develop?

-Benefits and issues.

What are the main benefits of UDA for the IS department? Are these benefits measured? If so, how? What are the main concerns of IS department about UDA?

-Success of UDA.

How is success of UDA defined by the IS department? Is UDA a success in the firm?

APPENDIX C
DP QUESTIONNAIRE

USER DEVELOPED COMPUTER-BASED APPLICATIONS
RESEARCH QUESTIONNAIRE
INFORMATION SYSTEMS

NOVEMBER 1981

SUZANNE RIVARD
SCHOOL OF BUSINESS ADMINISTRATION
THE UNIVERSITY OF WESTERN ONTARIO
(519) 679-6055

QUESTION A1.

The present study differentiates between three major classes of administrative computer-based applications. These classes are:

- a. Strategic planning applications which support the tasks of the managers who have to decide on the objectives of the organization, or changes in these objectives, and on the policies that are to govern the acquisition, use, and disposition of the resources. Applications assisting in economic forecasting, manpower planning, sales and profit planning would fall in this category.
- b. Management control applications, which assist the process by which managers assure that resources are obtained and used effectively in the accomplishment of the organization's objectives. Applications performing inventory control, demand forecasting, sales forecasting and analysis, capital requirements forecasting, cost analysis, and budgeting are instances of this type of applications.
- c. Operational control applications, which support the process of ensuring that specific tasks are carried out effectively and efficiently. This category includes transaction processing applications such as billing, accounts receivable, payroll, accounts payable, as well as applications such as plant scheduling, vehicle scheduling, order inquiry, and so on.

P20 Given these definitions, what is the approximate percentage of applications in each of the following categories?

	STRATEGIC PLANNING	MANAGEMENT CONTROL	OPERATIONAL CONTROL
(a) now being used	-----	-----	-----
(b) currently under development by IS	-----	-----	-----
(c) awaiting initiation of development	-----	-----	-----

QUESTION A2.

Is there a steering committee for Information Systems? _____

If yes,

(a) Who is the chairman? _____

(b) Who else is on it, by-title? _____

(c) What is the main function of the committee? _____

(d) How frequently does the committee meet? _____

(e) How effective is it in performing its main function?

1	2	3	4	5
Completely Ineffective				Extremely Effective

QUESTION A3.

If formality of planning is defined as the degree to which plans are in written form, how formal is the planning of Information Systems in your organization?

1	2	3	4	5
Very informal. No written plans other than per- haps the budget.		Plans written but not widely distributed throughout the organization.		Very formal. Detailed plans distributed wi- dely.

QUESTION A4.

What is the time span of the Information Systems plan? (check one)

- (a) One year _____
- (b) Two years _____
- (c) Five years _____
- (d) Ten years _____
- (e) Other (please specify) _____

QUESTION A5.

What role does the Information Systems plan play in guiding Informations Systems decisions?

1	2	3	4	5
No role.		Used sometimes.		No major decisions ma- de without re- ference to the plan.

QUESTION A6.

Do you charge users for:

	Yes	No
(a) System development?	_____	_____
(b) Data entry?	_____	_____
(c) CPU time?	_____	_____
(d) Terminal rental?	_____	_____
(e) Line charge to remote sites?	_____	_____
(f) Other? (Please specify)	_____	_____

QUESTION A7.

Are there written standards for programming and for systems development in use in your department?

	<input type="checkbox"/> Yes		<input type="checkbox"/> Yes
Systems Development		Programming	
	<input type="checkbox"/> No		<input type="checkbox"/> No

To what extent are these standards adhered to?

	1	2	3	4	5
Not at all					Completely

QUESTION A8.

Does your department make use of formal planning tools and/or formal control tools (such as PERT or CPM) for project development?

Yes

No

QUESTION A9.

Question on equipment.

QUESTION A10.

Is there in your department a group (or a person) who is responsible for data management?

Yes

No Please go to question A14.

If yes, how many data management staff do you have?

QUESTION A11.

To whom does the top data management person report in your organization?

- (a) Senior Information Systems Executive _____
 - (b) Director of Systems Development _____
 - (c) Other (Please specify) _____
-

QUESTION A12.

Which aspects of data management do your staff address?

- (a) Data Base design and development _____
 - (b) Data Base maintenance _____
 - (c) Logical data modeling _____
 - (d) Data analysis (assisting on system development teams, with the data-related aspects of the system) _____
 - (e) Information analysis (assisting users in determining their information needs by providing data base knowledge and skills) _____
 - (f) Other (Please describe) _____
-

QUESTION A13. Which of the following data management tools do you use?

- | | Name of tool |
|-----------------------------------|--------------|
| (a) Data Base Management Systems | _____ |
| (b) Data dictionary/
directory | _____ |
| (c) Query Language | _____ |
| (d) Report writer | _____ |
| (e) Other | _____ |

QUESTION A14.

When applications are developed by your department, what is the usual role of end users (check one):

- (a) Provide information to development staff (e.g. determination of outputs content, output formats) _____
- (b) Provide information as in (a) plus participate in the testing of the system _____
- (c) Participate to development as members of a development team _____
- (d) Other (Please specify) _____
- _____
- _____

QUESTION A15.

Which of the following are, or have been, problem areas for your department?

	Is a problem now	Was once a problem but no longer is	Exists but is not a problem	Has never occurred
(a) Large backlogs of applications	_____	_____	_____	_____
(b) End users departments renting computer time from outside organizations	_____	_____	_____	_____
(c) Size of maintenance load	_____	_____	_____	_____
(d) Pressures from end users to acquire own computers	_____	_____	_____	_____
(e) Non-traditional requirements from end users (e.g. model building, problem solving)	_____	_____	_____	_____
(f) Users want information more quickly than IS department can provide it	_____	_____	_____	_____
(g) Increasing programming costs	_____	_____	_____	_____
(h) Scarcity of qualified data processing personnel	_____	_____	_____	_____
(j) Users want to be able to access and manipulate their own data	_____	_____	_____	_____
(k) Other (Please specify)	_____	_____	_____	_____

SECTION B: USERS AS APPLICATIONS DEVELOPERS

DEFINITION 1

For the purposes of this questionnaire the word program is used as a general term to describe a set of instructions which a computer will execute. These instructions may be written using general purpose programming languages, statistical packages, data base query languages, modeling languages, report generators, etc.

DEFINITION 2

For the purposes of this questionnaire, the phrase application development activities includes tasks such as:

Development of New Computer-Based Applications

- a. Writing a program, using a query language (such as GIS, SEQUEL, NATURAL) to retrieve data from a data base.
- b. Writing a program, using a general purpose programming language (such as APL, BASIC, COBOL).
- c. Creating a program using a statistical package (such as SPSS, SAS) to analyze data.
- d. Writing programs, using a report generator (such as RPGII, EASYTRIEVE) to produce reports.
- e. Developing models, using a modeling language (such as IFPS, PLANCODE).
- f. Preparing graphics using a graphics package (such as SAS/GRAPH).

Applications Maintenance

Making changes in applications which already exist and which have been developed by the organization's Data Processing department, or by users. These changes may be required because the user's environment, tasks, or needs have changed.

CERTAIN QUESTIONS DEAL SPECIFICALLY WITH EITHER MAINTENANCE OR NEW APPLICATIONS DEVELOPMENT. IN THESE QUESTIONS, THE ACTIVITY WILL BE CLEARLY IDENTIFIED.

QUESTION B1.

Please estimate the number of end users in your organization who have developed or are developing administrative computer-based applications to support their job function.

QUESTION B2.

Of these users what percentage are members of:

- | | |
|-----------------------------------|---------|
| (a) Corporate Planning department | _____ % |
| (b) Finance department | _____ % |
| (c) Marketing department | _____ % |
| (d) Production department | _____ % |
| (e) Accounting | _____ % |
| (f) General Management | _____ % |
| (g) Personnel or Labour Relations | _____ % |
| (h) Others (Please specify) | _____ % |

QUESTION B3.

What is your best estimate of the percentage this number represents in proportion to the number of users you see as "potential" applications developers?

-----%

QUESTION B4.

Applications development done by end users is:

- (a) Mostly maintenance of applications
which have been developed by
the Data Processing department _____%
- (b) Mostly development of new appli-
cations _____%
- (c) Includes significant amounts of
both _____%

QUESTION B5.

Are these applications developed and run using:

- (a) A central computing facility
(e.g. Corporate computer) _____%
- (b) Outside organization <compu-
ting facilities _____%
- (c) Local computer (e.g. department
computer) _____%
- (d) Personal computer _____%
- (e) Other (Please specify) _____%

_____ 100 %

For the purposes of a number of the remaining questions the phrase PROGRAMMING TOOLS will be used as a synonym for programming languages such as BASIC, COBOL, APL, PL1, etc, software packages such as SPSS, MAPPER, IFPS, PLANCODE, and query languages such as QUERY-BY-EXAMPLE, SEQUEL, NATURAL.

QUESTION B6.

Are the programming tools users employ to develop applications:

- (a) Mostly provided by the Information Systems department _____
- (b) Mostly (or all) acquired by users from other sources _____

QUESTION B7.

Which, if any, programming tools does your department make available for users to use?

QUESTION B8.

Does your department provide users with support in the context of applications development?

- _____ Yes
- _____ No Go to QUESTION B11.

QUESTION B9.

If yes, what type of support do users receive?

- (a) Formal courses (or seminars) on the
use of programming aids _____
- (b) Consulting in the choice of the
proper programming tool _____
- (c) Consulting in the design of the
application _____
- (d) Debugging assistance _____
- (e) Space set aside for helping
and training users _____
- (f) Writing code _____
- (g) Other _____

QUESTION B10.

Does a separate organizational group whose responsibility is to provide end users with support, exist in your organization?

_____ Yes

_____ No

If no, who are the people who provide support to user?

GO TO QUESTION B11.

If yes to QUESTION B10., how many staff are there in that group?

Full time _____
Part time _____

To whom does the top person of this unit report in your organization?

- (a) Senior Information Systems Executive _____
- (b) Director of Systems Development _____
- (c) Other (Please specify) _____

QUESTION B11.

What are the main concern you had to face (or still have to face) in the context of users as applications developers?

QUESTION B12.

What benefits do you believe can be obtained by having users develop their own applications?

- (a) For the Information Systems departement?

(b) For the users?

SECTION C: CONTROL, MAINTENANCE & BACKLOGS

The following definition pertains to a number of questions in this section.

DEFINITION

For the purposes of this section we will differentiate between three types of administrative computer-based applications:

a. INDIVIDUAL SUPPORT APPLICATIONS

Applications designed to serve the personal information needs of one or a few individuals. These applications include queries, simple or sophisticated calculations, and other ad hoc data processing activities. They pertain to a task that is relatively independent of other tasks.

b. GROUP SUPPORT APPLICATIONS

These applications provide the reports, the queries, the analyses and the many other types of information required by a group of individuals engaged in interrelated tasks (for instance, an office, a department).

c. ORGANIZATIONAL SUPPORT APPLICATIONS

They are the applications which make up the central Data Processing activity of the organization. Inputs to these applications may come from several departments and the outputs used to plan or control the activities of several departments.

QUESTION C1.

From all the new administrative applications developed by end-users, what percentage are of:

- | | |
|------------------------------------|---------------|
| (a) Personal applications type | _____ % |
| (b) Departmental applications type | _____ % |
| (c) Corporate applications type | _____ % |
| | -----
100% |

QUESTION C2.

Are there any restrictions pertaining to the type of new applications users can develop?

_____ No. Go to QUESTION C3

_____ Yes.

If yes, which of the following are restrictions put forth in your organizations?

(a) No applications which would update files _____

(b) No applications which would imply the sharing of data between departments _____

(c) No applications which would involve access to one department's data by another department _____

(d) No applications beyond a certain size _____

(e) Others(Please specify)

What means are used to enforce these restrictions?

QUESTION C3.

Are there any controls put in place in the context of applications development by end-users (such as cost justification of applications, charge back for the use of computing facilities, access to data, etc...)?

No

Yes

If yes, what are these controls?

QUESTION C4.

What was the total of persons-hours expended annually on maintenance of applications before users started to develop applications?

The above answer is:

(a) Reasonably accurate, based on good data _____

(b) A rough estimate, based on minimal data _____

(c) A best guess, not based on any data _____

QUESTION C5.

On the total number of person-hours expended annually on maintenance at that time, what percentage was expended in each of the following areas?

- | | |
|--|---------|
| (a) Emergency program fixes, routine debugging | _____ % |
| (b) Accomodation of changes to data inputs and files, and accomodations of changes to hardware and system software | _____ % |
| (c) Enhancements requested by users | _____ % |
| (d) Improvement of program documentation | _____ % |
| (e) Recoding for efficiency in computation | _____ % |
| (f) Others(Please specify) | _____ % |
| _____ | _____ % |
| _____ | _____ % |
| | 100% |

QUESTION C6.

In the same period, in terms of the total person-hours worked annually by applications programming and systems analysis personnel, what percentage was spent in each of the following activities?

- | | |
|----------------------------------|---------|
| (a) Applications maintenance | _____ % |
| (b) New applications development | _____ % |
| (c) Other(Please specify) | _____ % |
| _____ | _____ % |
| _____ | _____ % |
| | 100% |

QUESTION C7.

Nowadays, what is the total of persons-hours expended annually on maintenance of applications?

The above answer is:

- (a) Reasonably accurate, based on good data _____
- (b) A rough estimate, based on minimal data _____
- (c) A best guess, not based on any data _____

QUESTION C8.

Nowadays, on the total number of person-hours expended annually on maintenance what percentage is expended in each of the following areas?

- (a) Emergency program fixes, routine debugging _____%
- (b) Accomodatio of changes to data inputs and files, and accomodations of changes to hardware and system software _____%
- (c) Enhancements requested by users _____%
- (d) Improvement of program documentation _____%
- (e) Recoding for efficiency in computation _____%
- (f) Others(Please specify) _____%

100%

QUESTION C9.

Nowadays, in terms of the total person-hours worked annually by applications programming and systems analysis personnel, what percentage is spent in each of the following activities?

(a) Applications maintenance _____%

(b) New applications development _____%

(c) Other(Please specify) _____%

100%

QUESTION C10.

Since users have started to develop applications, has your department

(a) Hired more programmers?

_____ Yes, how many? _____

_____ No

(b) Hired more systems analysts?

_____ Yes, how many? _____

_____ No

L

QUESTION C11.

Since users have started to develop new applications, were any of the following tools introduced for the use of applications programmers and systems analysts in your department?

(a) Data base dictionary/directory _____

(b) Decision tables _____

(c) Design aid techniques (e.g. SADT). _____

(d) Automated flowcharting _____

(e) Test data generators _____

(f) Structured walkthrough _____

(g) Chief programmer team _____

QUESTION C12.

If any of these tools were introduced, have you measured the resulting increase of productivity for applications programmers and systems analysts due to these tools?

_____ Yes, what was it? _____

_____ No, can you give a rough estimate of it? _____

APPENDIX D
CHECK-LIST FOR USER INTERVIEWS

CHECK-LIST FOR USER INTERVIEWS

COMPANY:
DATE:
USER:
USER TITLE:
DEPARTMENT:
DEPARTMENT RESPONSIBILITIES:

USER RESPONSIBILITIES:

BACKGROUND

FOR HOW LONG HAVE YOU BEEN DEVELOPING APPLICATIONS?
KNOWLEDGE OF DP BEFORE BEGINNING UDA?
FOR YOURSELF OR FOR OTHER PEOPLE?
IS IT A TASK WHICH TAKES MUCH OF YOUR WORKING-TIME?
NUMBER OF APPLICATIONS DEVELOPED?
WHY DEVELOP APPLICATIONS YOURSELF AND NOT ASK DP?
HAVE A TERMINAL IN THE OFFICE?

APPLICATIONS

LARGE OR SMALL APPLICATIONS? (TIME TO DEVELOP, TIME TO RUN)
IN GENERAL, APPLICATIONS WHO WILL BE USED FOR DECISION MAKING?
APPLICATIONS WHICH REPLACE SOMETHING WHICH WAS DONE PREVIOUSLY IN
ANOTHER WAY, OR DOING SOMETHING WHICH COULD NOT BE DONE PREVIOUSLY?
SOURCE OF DATA? IS ACCESS TO DATA A PROBLEM?
RUN REGULARLY OR ONE-SHOT? WHO RUNS APPLICATIONS?
WHAT TYPE OF APPLICATION? (MODELS, TRANSACTION PROCESSING, DATA
EXTRACTION, ETC....)
HAVE AN EXAMPLE OF A TYPICAL APPLICATION?

TOOLS

LANGUAGE(S) USED?
HOW DID YOU LEARN THE LANGUAGES(S)?
FOR EACH LANGUAGE: EASY TO LEARN? TO USE? REQUIRES PRACTICE?
FLEXIBLE?

SUPPORT

RELY ON DP (INFORMATION CENTRE, TIMESHARING SERVICES, ETC.) FOR SUPPORT?

WHAT IS IMPORTANT WITH RESPECT TO SUPPORT?

WHAT ABOUT SUPPORT FROM (OR TO) OTHER USERS?

ASSESSMENT OF SUPPORT PROVIDED BY DP.

ECONOMICS OF UDA

PREPARE FORMAL COST BENEFITS ANALYSIS?

(YES) WHY? DOES SOMEBODY ASK FOR IT?

(NO) WHY?

IF DEVELOPS APPLICATIONS FOR OTHER PEOPLE, DO THOSE PEOPLE HAVE TO PREPARE FORMAL COST BENEFITS ANALYSIS?

HOW IS THE DECISION MADE TO DEVELOP AN APPLICATION?

ARE THERE PRODUCTIVITY IMPROVEMENTS? ARE THEY CALCULATED?

IMPROVEMENT IN DECISION MAKING?

DOES DP CHARGE USERS FOR USE OF COMPUTER RESOURCES?

(IF YES) DOES IT INFLUENCE THEIR USE OR RESOURCES?

MISCELLANEOUS

MAIN ADVANTAGES OF UDA?

MAIN PROBLEMS HAVE TO FACE?

WHAT DOES SUPERIOR THINK ABOUT UDA?

IS HE/SHE INVOLVED?

COMMENTS, SUGGESTIONS, ETC.

APPENDIX E

SURVEY QUESTIONNAIRE & COVERING LETTER

RESEARCH PROJECT ON USER DEVELOPED COMPUTER-BASED APPLICATIONS

A team of investigators, at the School of Business Administration, the University of Western Ontario is conducting an ongoing research to investigate the management of the Information Systems function. Suzanne Rivard, a Ph.D. candidate at the School, is a member of this team. She is conducting a study to investigate situations where Data Processing users develop their own administrative computer-based applications. This study constitutes her thesis and is a major requirement for earning her degree.

Users who develop applications are a critical source of information for this research project. The attached questionnaire was designed to gather detailed data from such users. We are grateful to the (DATA PROCESSING DEPARTMENT OR INFORMATION CENTRE) of (COMPANY NAME) for its contribution to the study by providing us with a list of their users and to (COMPANY NAME) itself for allowing us to use its internal mail system to distribute the questionnaires.

We would very much appreciate your assistance in our study. Would you please complete this questionnaire and return it directly to us as soon as your schedule permits. A pre-addressed envelope is attached for returning the questionnaire. You may be assured of complete confidentiality. Your answers will be combined with answers from users of several other firms, and information gathered from this questionnaire will be revealed in aggregate form only.

May we stress the importance of timing. The schedule set for Ms. Rivard's research project calls for the completion of this phase within a few weeks.

We thank you for your assistance in this study. If you should need any further information do not hesitate to contact Suzanne Rivard, at (519) 679-6055.

Sincerely,

Sid Huff
Professor of Business Administration



The University of Western Ontario
School of Business Administration

USER DEVELOPED COMPUTER-BASED APPLICATIONS

Research Questionnaire

IT SHOULD TAKE ABOUT 30 MINUTES OF YOUR TIME TO COMPLETE THIS QUESTIONNAIRE. THE QUESTIONS CAN BE ANSWERED VERY QUICKLY AND WILL NOT REQUIRE DETAILED NUMBERS OR RECORDS. THANK YOU FOR YOUR HELP.

SUZANNE RIVARD
DOCTORAL PROGRAM RESEARCH
SCHOOL OF BUSINESS ADMINISTRATION
THE UNIVERSITY OF WESTERN ONTARIO
LONDON, ONTARIO, N6A 3K7

(519) 679-6055 — OFFICE
(519) 472-8075 — HOME

SECTION I: DEFINITIONS

PLEASE NOTE

This section includes definitions of terms which will be used for the purposes of the present study. Please read them carefully; this will permit you to complete the questionnaire faster and more accurately.

DEFINITION 1

The phrase "organization's Data Processing department" is used as a synonym for terms such as Information Systems department, Computer Services, Computing Resources department, and so on.

DEFINITION 2

For the purposes of this questionnaire, the word PROGRAM is used as a general term to describe a set of instructions which a computer will execute. These instructions may be written using general purpose programming languages, statistical packages, data base query languages, modeling languages, report generators, etc.

DEFINITION 3

For the purposes of this questionnaire, the phrase PROGRAMMING TOOLS will be used as a synonym for general purpose programming languages such as APL, BASIC, COBOL, FORTRAN, etc., data base query languages such as GIS, SEQUEL, NATURAL, etc., software packages such as GENMOD, IFPS, MAPPER, PLANCODE, SAS, SPSS, etc., report generators such as RPGII, EASYTRIEVE, and so on.

DEFINITION 4

For the purposes of this questionnaire, the phrase application development activities includes tasks such as:

Development of New Computer-Based Applications

- a. Creating programs, using a query language (such as GIS, SEQUEL, NATURAL) to retrieve data from a data base.
- b. Creating programs, using a general purpose programming language (such as APL, BASIC, COBOL).
- c. Creating programs using a statistical package (such as SPSS, SAS) to analyze data.
- d. Creating programs, using a report generator (such as RPGII, EASYTRIEVE) to produce reports.
- e. Developing models, using a modeling language (such as IFPS, PLANCODE).
- f. Preparing graphics, using a graphics package (such as SAS/GRAPH).

Applications Maintenance

Making changes in applications which already exist. These changes may be required because the user's environment, tasks, or needs have changed.

CERTAIN QUESTIONS DEAL SPECIFICALLY WITH EITHER MAINTENANCE OR NEW APPLICATIONS DEVELOPMENT. IN THESE QUESTIONS, THE ACTIVITY WILL BE CLEARLY IDENTIFIED.

SECTION II: GENERAL QUESTIONS ABOUT APPLICATION DEVELOPMENT

Q 1. Of the total number of hours spent at work (during a normal week or over a year) what percentage is spent doing application development?

Q 2. Of the time you spend performing application development activities, what percentage is devoted to each of the following categories of activities?

Please indicate the PROGRAMMING TOOL you use the most often for each category.

	PERCENTAGE	NAME OF TOOL MOST OFTEN USED
a. Developing models, the parameters of which will be changed to answer 'what if' type of questions	_____ %	_____
b. Creating programs (or parts of programs) which will update files	_____ %	_____
c. Creating programs (or parts of programs) which will perform data analysis (e.g. contingency tables, regression analysis)	_____ %	_____
d. Creating programs (or parts of programs) which retrieve data from a file or a data base, perform some calculations on the data, and produce reports	_____ %	_____
e. Creating programs (or parts of programs) which retrieve data from a file or a data base and produce a list or report, without performing any calculations	_____ %	_____
f. Preparing graphics, using a graphics package	_____ %	_____
g. Other (Please specify) _____	_____ %	_____
	_____ %	_____
	100%	

Q 3. Of the time you spend performing application development activities, what percentage is devoted to develop applications in each of the following categories?

- a. **INDIVIDUAL SUPPORT APPLICATIONS:**
Applications designed to serve the personal information needs of one or a few individuals. They pertain to a task that is relatively independent of other tasks.
- b. **GROUP SUPPORT APPLICATIONS:**
These applications provide the reports, the queries, the analyses and the many other types of information required by a group of individuals engaged in interrelated tasks (for instance an office, a department).
- c. **ORGANIZATIONAL SUPPORT APPLICATIONS:**
They are applications the inputs of which may come from several departments, and the outputs of which are used to plan or control the activities of several departments.

Q 4. Of all your application development activities, what percentage do you perform:

- a. As a member of a team (2 or more) of users? _____ %
 - b. By yourself? _____ %
 - c. Other? (Please specify) _____ %
- _____ %
- _____ %
- 100%

Q 5. Of all your application development activities involving the use of a computer, what percentage are performed using:

- a. The company's central computer _____ %
 - b. Another company's computer _____ %
 - c. A computer within your department _____ %
 - d. A personal computer (e.g. APPLE) _____ %
 - e. Other (Please specify) _____ %
- _____ %
- 100%

Q 6. A. Of all the hours you spend using the computer, either to develop applications or run applications you developed, what percentage fall in each of the following categories?

- a. Have the dedicated use of a terminal in the office _____ %
 - b. Share the use of terminal(s) with others in your department _____ %
 - c. Use terminals which are located in an Information Centre _____ %
 - d. Use terminals within the Data Processing department, other than in an Information Centre _____ %
 - e. Use a home terminal _____ %
 - f. Other (Please specify) _____ %
- _____ %
- 100%

B. Are there any applications that you develop and run without using a terminal (e.g. punched cards, filling out forms which are transmitted to DP)?

- ___ Yes
- ___ No

Q 7. Did you have the same job in this organization at the time you started developing applications?

- ___ Yes
- ___ No, I was _____ in this organization.
Job Title
- ___ No, I was _____ elsewhere.
Job Title

Q 8. Which of the following best describes your knowledge of computers and programming, before you started developing applications in this organization? (Please check one)

- a. Absolutely no knowledge of computer programming and of computer use. _____
- b. No knowledge of computer programming but had previously used computers (for data entry or for running existing applications, for instance) _____
- c. Some knowledge of computer programming (had taken one or two courses, for instance) _____
- d. Very good knowledge of computer programming (either due to work experience or/and a degree in a computing-intensive area of study) _____

Q 9. For how long have you been doing application development in this organization?

_____ months or _____ years

Q 10. During the period referred to in QUESTION 9, please estimate the number of new applications (as they are defined in SECTION 1) you developed.

_____ new applications. If none, please go to QUESTION 13.

Q 11. A. Do you develop applications for other people than yourself and/or the individual to whom you report?

___ Yes

___ No Please go to QUESTION 12.

B. If yes, what percentage of the applications you have developed were developed for other people than yourself or/and the individual to whom you report?

_____ %

C. Is it part of your formal duties to develop applications for other people than yourself or/and the individual to whom you report?

___ Yes

___ No

Q 12. A. What percentage of the new applications you have developed took:

- a. Less than one day to develop? _____ %
- b. One day to one week to develop? _____ %
- c. One to two weeks to develop? _____ %
- d. Two weeks to one month to develop? _____ %
- e. One to three months to develop? _____ %
- f. More than three months to develop? _____ %

100%

B. What percentage of the applications you have developed:

a. Use files that you created by inputting data (e.g. from reports, documents)? _____

b. Use files that you created from already existing files or data bases made available by the Data Processing department? _____

c. Use already existing files or data bases made available by the data processing department? _____

d. Other? (Please specify) _____

100%

C. Of the new applications you have developed, what percentage falls in each of the following categories?

a. One shot type of applications _____

b. Applications which are run on a regular basis (e.g. monthly, weekly, daily) _____

c. Other (Please specify) _____

100%

D. What percentage of the applications you have developed:

a. Do you run yourself? _____

b. Does your superior(s) run? _____

c. Do your colleagues run? _____

d. Do your subordinate(s) run? _____

e. Other? (Please specify) _____

100%

E. The following statements describe the use which can be made of the information provided by the applications you have developed. Please indicate the percentage of applications which fall in each category.

a. You use the information as a basis of group work with colleagues _____

b. You personally make use of the information provided by the applications you have developed _____

c. You directly transmit the information to the individual to whom you report _____

d. You directly transmit the information to other people than yourself or the individual to whom you report _____

e. The information is used by you to formulate a suggested action which is then presented to the individual to whom you report _____

f. Other (Please specify) _____

100%

Q 13. A. Do you do any maintenance (as defined in SECTION I) of applications which have been developed by the Data Processing department?

___ Yes

___ No. (Please go to QUESTION 14.)

B. Please consider all your requirements for maintenance of applications developed by the Data Processing department. For what percentage of these requirements do you do maintenance yourself?

Q 14. What in your opinion is the major advantage of developing applications yourself rather than having the Data Processing department do the work?

Q 15. What in your opinion is the major disadvantage of developing applications yourself rather than having the Data Processing department do the work?

Q 16. A. Please consider the "one shot" application which took you the longest period of time to develop (if none, please go to QUESTION 17.) How would you compare the time it took to develop and run this application to the time it used to take or would have taken to accomplish the same task otherwise (e.g. using pencil and paper, calculator, etc.)?

TO ACCOMPLISH THE SAME TASK WITHOUT THE APPLICATION WOULD TAKE:

DO NOT KNOW SAME TIME TWICE AS MUCH 3 TIMES MORE 4 TIMES MORE 5 TIMES MORE 6 TIMES MORE 7 TIMES MORE 8 TIMES MORE

B. Please consider the "one shot" application which took you the less time to develop (if none please go to QUESTION 17). How would you compare the time it took you to develop and run this application to the time it used to take or would have taken to accomplish the same activity otherwise (e.g. using pencil and paper, calculator, etc.)?

TO ACCOMPLISH THE SAME TASK WITHOUT THE APPLICATION WOULD TAKE:

DO NOT KNOW SAME TIME TWICE AS MUCH 3 TIMES MORE 4 TIMES MORE 5 TIMES MORE 6 TIMES MORE 7 TIMES MORE 8 TIMES MORE

Q 17. Please consider all the applications you have developed and which are run on a regular basis (if none, please go to QUESTION 18).

A. On the average how long does it take you to develop this type of applications?

_____ HOURS DAYS

B. On the average how would you compare the time it takes to run such applications (including the time to input the data, when it applies) to the time it used to take or would have taken to accomplish the same activity otherwise (e.g. using pencil and paper, calculator, etc.)?

TO ACCOMPLISH THE SAME TASK WITHOUT THE APPLICATION WOULD TAKE:

DO NOT KNOW SAME TIME TWICE AS MUCH 3 TIMES MORE 4 TIMES MORE 5 TIMES MORE 6 TIMES MORE 7 TIMES MORE 8 TIMES MORE

Q 18. Please circle one number to indicate to what extent you think the applications you have developed provide benefits due to better decisions made on the basis of the information provided by the applications.

1	2	3	4
-----	-----	-----	-----
Is not a benefit.	May be a benefit, but has never received much attention.	It is strongly felt that it is a benefit. Yet its monetary value has never been evaluated.	It is a benefit and there are numbers to support it.

SECTION III: EVALUATION OF THE CONTEXT OF APPLICATION DEVELOPMENT

REMINDER: The phrase PROGRAMMING TOOLS is used as a synonym for general purpose programming languages such as APL, BASIC, COBOL, FORTRAN, etc., data base query languages such as GIS, SEQUEL, NATURAL, etc., software packages such as GENMOD, IPFS, MAPPER, PLANCODE, SAS, SPSS, etc., report generators such as RPGII, EASYTRIEVE, and so on.

Q 1. This question asks you to indicate the importance of each of several aspects of application development by users.

Please place an X on each line to indicate how important each aspect is to you. There are no right or wrong answers. Only your opinion is important.

An example is provided for a person who thought that availability of terminals was somewhat important.

EXAMPLE Availability of terminals is Important: ___ : ___ : ___ : X : ___ : Unimportant

FOR MYSELF

Ease of use of the programming tools is Important: ___ : ___ : ___ : ___ : Unimportant

To be aware of the programming tools available is Important: ___ : ___ : ___ : ___ : Unimportant

Quality of training with the use of programming tools is Important: ___ : ___ : ___ : ___ : Unimportant

Assistance in the choice of the proper programming tool for a given application is Important: ___ : ___ : ___ : ___ : Unimportant

Assistance during the design (i.e. steps prior to actually writing programs) is Important: ___ : ___ : ___ : ___ : Unimportant

Assistance in finding out and fixing errors in a program is Important: ___ : ___ : ___ : ___ : Unimportant

Assistance with technical aspects (e.g. setting up files) is Important: ___ : ___ : ___ : ___ : Unimportant

Good knowledge of the programming tools by those who provide assistance to users is Important: ___ : ___ : ___ : ___ : Unimportant

Understanding of the applications I develop by those who provide assistance to users is Important: ___ : ___ : ___ : ___ : Unimportant

"People orientation" of those who provide assistance to users is Important: ___ : ___ : ___ : ___ : Unimportant

Availability of those who provide assistance to users is Important: ___ : ___ : ___ : ___ : Unimportant

Fast response time when I run an application is Important: ___ : ___ : ___ : ___ : Unimportant

Availability of terminals is Important: ___ : ___ : ___ : ___ : Unimportant

Not having to explain to a DP professional what an application should do is Important: ___ : ___ : ___ : ___ : Unimportant

To have an application developed in a more timely fashion than if it was developed by Data Processing is Important: ___ : ___ : ___ : ___ : Unimportant

To have control over the applications I develop is Important: ___ : ___ : ___ : ___ : Unimportant

To have applications developed at a lower cost than if they were developed by Data Processing is Important: ___ : ___ : ___ : ___ : Unimportant

For supporting me in my job, the information provided by the applications I develop is Important: ___ : ___ : ___ : ___ : Unimportant

Q 2. Does the Data Processing department provide (either directly or through an Information Centre, User Support Group, etc.) the following types of assistance to users who develop applications?

	YES	NO	DO NOT KNOW
a. Formal courses or individual training on the use of the programming tools	___	___	___
b. Assistance in the choice of the proper programming tool for a given application	___	___	___
c. Assistance in the design (i.e. steps prior to actually writing the programs) of applications	___	___	___
d. Assistance in finding out and fixing errors in programs	___	___	___
e. Technical assistance (e.g. setting up files)	___	___	___

Q 3. How, in this organization, did you learn to use programming tools? (If several categories apply, please rank them, letting 1 be the most important, 2 the second most important, and so on.)

- a. Alone, using books, user manuals, etc. _____
- b. With the assistance of the Data Processing department (including Information Centre, User Support Group, etc.) _____
- c. With the assistance of outside consultants _____
- d. With the assistance of vendors _____
- e. With the assistance of colleagues _____
- f. Other (Please specify) _____

Q 4. Please circle one number to indicate how often you received each type of assistance from the Data Processing department (either directly or through an Information Centre, User Support Group, etc.).

	NEVER	OCCASIONALLY	OFTEN	VERY OFTEN
Assistance in the choice of the proper programming tool to use for a given application	1	2	3	4
Assistance in the design (i.e. the steps prior to actually writing programs) of applications	1	2	3	4
Assistance in finding out and fixing errors in the programs you write	1	2	3	4
Technical assistance (e.g. setting up files)	1	2	3	4

5. Please circle one number on each line to indicate the extent to which you are satisfied or dissatisfied with each of the following aspects of application development.

	VERY SATISFIED	SATISFIED	NEUTRAL	DISSATISFIED	VERY DISSATISFIED
Ease of use of the programming tools	1	2	3	4	5
Your awareness of the programming tools available	1	2	3	4	5
Quality of training (with the use of programming tools) you have received	1	2	3	4	5
Assistance provided by Data Processing in the choice of the proper programming tool for a given application	1	2	3	4	5
Assistance provided by Data Processing during the design of applications.	1	2	3	4	5
Assistance provided by Data Processing in finding out and fixing errors in programs you write	1	2	3	4	5
Technical assistance provided by Data Processing	1	2	3	4	5
Knowledge of the programming tools by those who provide assistance to users	1	2	3	4	5
Understanding of the applications you develop by those who provide assistance to users	1	2	3	4	5
"People orientation" of those who provide assistance to users	1	2	3	4	5
Availability of those who provide assistance to users	1	2	3	4	5
Response time when you run an application	1	2	3	4	5
Availability of terminals	1	2	3	4	5
Not having to explain to a DP professional what an application should do	1	2	3	4	5
Reduction of development cost due to the fact that you develop the applications rather than asking Data Processing to develop them	1	2	3	4	5
The timeliness with which applications are developed (as compared to having them developed by Data Processing)	1	2	3	4	5
The control you have over the applications you develop	1	2	3	4	5
The usefulness of the information provided by the applications you have developed, in supporting you in your job	1	2	3	4	5

Q 6. Overall, taking everything into consideration, how satisfied are you with the experience of developing applications in this organization?

EXTREMELY SATISFIED	SOMEWHAT SATISFIED	SATISFIED	NEUTRAL	DISSATISFIED	SOMEWHAT DISSATISFIED	EXTREMELY DISSATISFIED
1	2	3	4	5	6	7

Q 7. Below is a series of statements about application development by users. Please circle one number on each line to indicate how much you agree or disagree with each statement. If a statement does not apply, please leave the line blank. There are no right or wrong answers. Only your opinion is important.

	STRONGLY AGREE	AGREE	NEUTRAL	DISAGREE	STRONGLY DISAGREE
The idea of application development by users is welcome by people in my department	1	2	3	4	5
Colleagues are a better source of support than the Data Processing department people	1	2	3	4	5
The Data Processing department is at odds with users developing their own applications	1	2	3	4	5
I feel that I have the expertise required to develop the applications I do develop	1	2	3	4	5
Data processing heavily promotes the idea of application development by users	1	2	3	4	5
Applications users develop themselves are often more useful than applications developed by Data Processing	1	2	3	4	5
For me, developing applications is a waste of time	1	2	3	4	5
In this firm, the major thrust toward application development by users comes from the users community	1	2	3	4	5
Data Processing has too much control over the applications users develop	1	2	3	4	5
I think it is not the job of users to develop applications	1	2	3	4	5
When users develop applications themselves, they better understand why applications developed by the Data Processing department often take longer than planned	1	2	3	4	5
It is difficult to develop applications	1	2	3	4	5
The Data Processing department tries to facilitate access to the computer to users who want to develop applications	1	2	3	4	5
I have often asked Data Processing to provide facilities (e.g. programming tools, training, assistance) to users who want to develop applications (either for myself or for other users)	1	2	3	4	5
People in my department (other than myself) have often asked Data Processing to provide facilities to users who want to develop applications	1	2	3	4	5

Q 8. A. On the scale below, please circle one number to indicate how much the person to whom you report encourages the fact that you develop applications.

1	2	3	4	5	6

IMPOSES IT	STRONGLY ENCOURAGES	ENCOURAGES	IS INDIFFERENT	DISCOURAGES	STRONGLY DISCOURAGES

B. On the scale below, please circle one number to indicate the degree to which you think the person to whom you report is involved in application development.

1	2	3	4	5

NOT INVOLVED AT ALL				VERY MUCH INVOLVED (E.G DEVELOPS APPLICATIONS)

SECTION IV: EVALUATION OF THE PROGRAMMING TOOLS

PLEASE NOTE: In this section, you are asked to evaluate the programming tools you use. Since some people may use more than one programming tool, the section includes the same set of statements in order to evaluate up to two programming tools.

Q 1. This question pertains to the programming tool you use the most often.

A. Please give the name of this tool:

B. Please place an X on each of the five point scale to indicate how you feel about each statement. When a statement does not apply to your situation, please check the 'DOES NOT APPLY' column.

An example is provided for a person who thought that keywords of the programming tools used were very easy to remember.

EXAMPLE:

The keywords are: Easy to remember : X : : : : : Hard to remember

	DOES NOT APPLY
For you, learning how to use this tool took	A very short period of time : : : : : A very long period of time : _____
In order to use this tool	Few keywords are required : : : : : Many keywords are required : _____
You have found these keywords	Easy to remember : : : : : Hard to remember : _____
To you these keywords are	Meaningful : : : : : Meaningless : _____
When you want to modify a program, the code is	Easy to modify : : : : : Hard to modify : _____

This programming tool can be used for	Many different types of tasks	: _____ : _____ : _____ : _____ : _____	One type of task	_____
The format in which you have to write a program is	Flexible (keywords can be anywhere on a line)	: _____ : _____ : _____ : _____ : _____	Rigid (e.g. specific columns for specific parts of a statement)	_____
When there is an error in a program, the error messages are	Easy to understand	: _____ : _____ : _____ : _____ : _____	Hard to understand	_____
You find the error messages	Helpful in fixing errors	: _____ : _____ : _____ : _____ : _____	Not helpful in fixing errors	_____
Do you feel that	The smallest error (e.g. a missing comma or an extra space) aborts a run	: _____ : _____ : _____ : _____ : _____	Only severe errors will cause the program to stop running	_____
The "help" commands on the system	Are indeed helpful	: _____ : _____ : _____ : _____ : _____	Are not helpful	_____
The reference manual (or user manual) is	Helpful	: _____ : _____ : _____ : _____ : _____	Not helpful	_____
Your background in computer programming was	Sufficient to effectively use the tool	: _____ : _____ : _____ : _____ : _____	Insufficient to effectively use the tool	_____
To have the computer do a simple job	A few instructions are sufficient	: _____ : _____ : _____ : _____ : _____	Several instructions are required	_____
In your knowledge, / this tool could handle	Large and complex types of jobs	: _____ : _____ : _____ : _____ : _____	Small and simple types of jobs	_____
For all the types of applications you develop, you use	A single programming tool	: _____ : _____ : _____ : _____ : _____	Many programming tools	_____

If you use only one programming tool, please go directly to SECTION V: BASIC-BACKGROUND INFORMATION.

Q 2. This question pertains to the second most often used programming tool.

A. Please give the name of this tool: _____

B. Please place an X on each of the five point scale to indicate how you feel about each statement. When a statement does not apply to your situation, please indicate so in the 'DOES NOT APPLY' column.

								DOES NOT APPLY	
For you, learning how to use this tool took	A very short period of time	: _ : _ : _ : _ : _	:	:	:	:	:	A very long period of time	___
In order to use this tool	Few keywords are required	: _ : _ : _ : _ : _	:	:	:	:	:	Many keywords are required	___
You have found these keywords	Easy to remember	: _ : _ : _ : _ : _	:	:	:	:	:	Hard to remember	___
To you these keywords are	Meaningful	: _ : _ : _ : _ : _	:	:	:	:	:	Meaningless	___
When you want to modify a program, the code is	Easy to modify	: _ : _ : _ : _ : _	:	:	:	:	:	Hard to modify	___
This programming tool can be used for	Many different types of tasks	: _ : _ : _ : _ : _	:	:	:	:	:	One type of task	___
The format in which you have to write a program is	Flexible (keywords can be anywhere on a line)	: _ : _ : _ : _ : _	:	:	:	:	:	Rigid (e.g. specific columns for specific parts of a statement)	___
When there is an error in a program, the error messages are	Easy to understand	: _ : _ : _ : _ : _	:	:	:	:	:	Hard to understand	___
You find the error messages	Helpful in fixing errors	: _ : _ : _ : _ : _	:	:	:	:	:	Not helpful in fixing errors	___
Do you feel that	The smallest error (e.g. a missing comma or an extra space) aborts a run	: _ : _ : _ : _ : _	:	:	:	:	:	Only severe errors will cause the program to stop running	___
The "help" commands on the system	Are indeed helpful	: _ : _ : _ : _ : _	:	:	:	:	:	Are not helpful	___
The reference manual (or user manual) is	Helpful	: _ : _ : _ : _ : _	:	:	:	:	:	Not helpful	___
Your background in computer programming was	Sufficient to effectively use the tool	: _ : _ : _ : _ : _	:	:	:	:	:	Insufficient to effectively use the tool	___
To have the computer do a simple job	A few instructions are sufficient	: _ : _ : _ : _ : _	:	:	:	:	:	Several instructions are required	___
In your knowledge, this tool could handle	Large and complex types of jobs	: _ : _ : _ : _ : _	:	:	:	:	:	Small and simple types of jobs	___
For all the types of applications, you develop, you use	A single programming tool	: _ : _ : _ : _ : _	:	:	:	:	:	Many programming tools	___

SECTION V: BASIC DEMOGRAPHIC-BACKGROUND INFORMATION

Q 1. A. Name of your company: _____

B. Your business could best be described as: (Please check one)

- | | | | |
|---|-------|---------------------------|-------|
| Manufacturing consumer durable products | _____ | Defense Industry | _____ |
| Manufacturing consumer non-durable products | _____ | Social Services | _____ |
| Manufacturing industrial products | _____ | Government | _____ |
| Banking | _____ | Retail or wholesale trade | _____ |
| Investment | _____ | Transportation | _____ |
| Insurance | _____ | Utility | _____ |
| Construction | _____ | Other (please specify) | _____ |
| Mining | _____ | | |
| Oil | _____ | | |

Q 2. Which of the following best describes the functional area in which you work?

- | | | | |
|--------------------|-------|------------------------------|-------|
| Accounting | _____ | Marketing | _____ |
| Engineering | _____ | Personnel or Labor Relations | _____ |
| Finance | _____ | | |
| General Management | _____ | Other (please specify) | _____ |
| Production | _____ | | |

Q 3. Your present title or position is: _____

Thank you very much for filling out this questionnaire. If there is anything else you would like to mention about your experience of developing computer based applications, please use the back of this page for that purpose. These comments will be most appreciated.

APPENDIX F
SCENARIOS OF UDA

COMPANY A

1. The Company

A is a financial institution, employing 23,000 people, and with assets in excess of \$50 billion. The largest portion of the company's activities take place in Canada, but A also does business in more than 100 countries.

2. DP Profile2.1. Basic Facts

A has been using computers for commercial DP for more than 20 years. For many years, DP had been the major initiator of projects: DP professionals would see opportunities due to technological developments and could see how to apply them to the firm's activities. Users were not very much involved with DP, and a gap, both mental and physical was deepening between users and DP. The mental gap, which was due to the very different profile of DP professionals from other people in the firm, was worsened by physical separation: DP activities are performed in different locations than the head office. Given that situation, a few years ago, top management stipulated that the mandate of DP was twofold: make DP become part of the culture of the company while optimizing the operations of the firm, because of increasing costs. Nowadays, DP employs approximately 1500 people. Company A has two central data centres, and nine regional data centres. At A, computer power is provided by two IBM 3033's, one IBM 3031, and one IBM

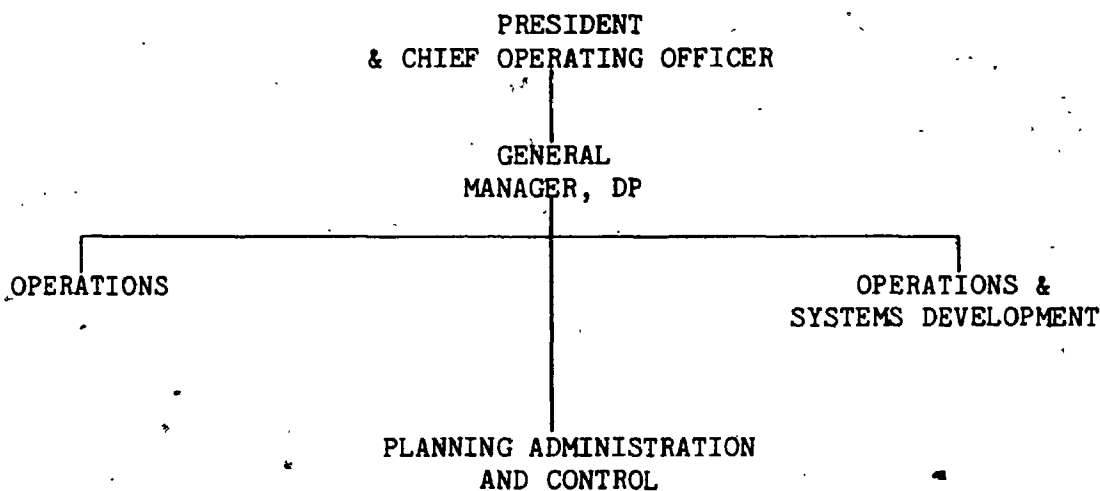
370-168, and one Amdhal V7. No major processing is performed at the regional data centres. Their main role is to be "large, sophisticated RJE stations." DP's monthly rental-equivalent budget for hardware and communications is in excess of \$2.5 million.

2.2 DP Organization

The General Manager in charge of DP reports directly to the President and Chief Operating Officer of the company. Three departments constitute the DP group; they are Operations Processing, Planning Administration and Control, and Operations and Systems Development (see Figure A.1).

Operations Processing is responsible for the management of the data centres, for providing computer services, and for office processing which includes stationery, printing, copy centre, and so on. Planning, Administration, and Control provides three main types of services. First, the group provides administrative services to the DP community. Such services are personnel, facilities and security, traffic (coordination of data transmission), as well as word processing. Second, a group is responsible for planning and control. The planning activity mostly consists of integrating the plans provided by the Operations department and by the Operations and Systems Development department. The control group is mainly responsible for ensuring that the systems developed and implemented respect the company's security requirements. The third group provides services to both the user community and to DP. The group is responsible for time and motion studies, for helping to optimize office operations that are

FIGURE A.1
DP ORGANIZATION AT A

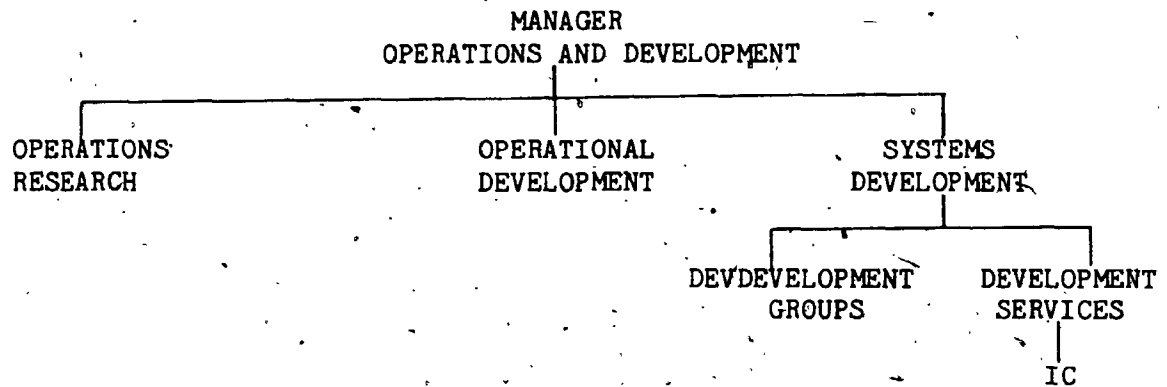


not computerized, and for providing "test environments" wherein new systems are experimented with before being implemented in the "company at large."

Operations and Systems Development includes three groups, namely, Operations Research, Operational Development, and Systems Development, within which the Information Centre (IC) is located (see Figure A.2). Operations Research is responsible for developing operating models of the firm; some members of the group work in user departments in order to identify areas where OR can be applied. Operational Development is responsible for performing pre-feasibility studies ("concept analysis") as well as for testing the applications developed by Systems Development. The group is also responsible for some prototyping.

FIGURE A.2

OPERATIONS AND SYSTEMS DEVELOPMENT



Systems Development is responsible for the development of information systems for users. Within this department, Development Services is responsible for providing services and standards to the development group (such as developing systems development methodologies). The IC is part of this Development Services group.

2.3 DP from the Inside

It is estimated that operational control systems account for approximately 85 percent of the DP budget, management control systems for 10 percent, and strategic planning for 5 percent. An in-house developed systems development methodology is used, and standards for programming are also in place. The concept of Data Administration is not implemented, but the data base technology is in place. The data base management system at A is IMS. According to several people in the company (both from DP and from the user community) the movement toward

Data Administration is slowed down by issues regarding ownership as well as confidentiality and security of data.

A 5 year plan and a 2 year plan are prepared by DP every year. However, these plans are operational plans rather than strategic plans. As put by a manager within DP: "DP does not do real planning -A is not planning itself."

2.4 Relationships with the User Community

A Resource Allocation Steering Committee exists, which is chaired by the Chief Financial Officer of the firm, and consists of by representatives of the user community, at the level of general managers. The main responsibility of the committee is to approve the allocation of resources: business cases are presented and the committee gives (or refuses) its approval. The committee meets approximately every two months.

DP is centralized at A. However, some departments (very few of them) have mini-computers devoted to stand-alone systems. For the last three years, DP has charged users for systems development; the rate is \$250 per day. Users are not charged for other DP services, but they receive reports indicating how much computer time they use. However, user departments which have terminals are responsible for those terminals' costs.

3. UDA at A

3.1. Background

At A, an IC has been in place for three and a half years. Its implementation was motivated by the fact that the application backlog was very large, and was composed, for a large proportion, of requests for extracting and "massaging" data. On the one hand, users were dissatisfied because they had to wait long for such applications to be developed. On the other hand, DP professionals were also upset: their skills and interests are oriented more toward the development of large and complex production systems rather than toward small, one-shot type of applications. Providing users with tools that would enable them to develop the type of applications mentioned above was seen as a means of both reducing the pressure on DP and making users more satisfied. A business case was prepared for the IC and was approved by top management of A.

The initial manager of the IC had a marketing orientation: a major objective was to make the IC known, and to attract new users. The number of users grew very rapidly, from approximately 30 at the end of 1979 to more than 150 less than two years later. In addition to growing rapidly, UDA also grew in a fashion different from what DP expected. That is, while many of the applications that users develop extract and manipulate data, several others are of transaction processing type, where files are created, updated, and reports prepared.

3.2 UDA Now

Nowadays, one manager and six consultants staff the IC. The present manager replaced the initial manager who left approximately six months ago. The mandate of the new manager is to "control IC usage and keep users happy," which implies halting IC promotion efforts, improving user support, and evaluating different chargeout schemes. In the very near future, IC users will be charged for their use of DP resources, on the basis of CPU time plus IC overhead.

While most DP activities are performed in locations which are remote from the head office, the IC is located at the head office. There is a walk-in terminal room, with 8 terminals and one printer, as well as a walk-in consulting room. Approximately 20 terminals are located in user departments, for the purpose of using the IC services. The main programming tools offered to users are GIS and SAS, which both operate in a batch mode. For each tool, one day seminars are offered, every one or two months for GIS, once or twice a month for SAS. Other tools are available to users, but are not as much used as the two previous ones. Moreover, to date, the IC does not provide training with the use of these tools, which are: ADRS-II, APL, BASIC, and FPS. According to a senior IC consultant, usage of ADRS-II and of APL is low, and is likely to remain as such [1]. Usage of BASIC is low but increasing while FPS usage is low and decreasing, to be replaced by FPS.

Responsibilities of the IC staff include training, technical assistance, debugging assistance, as well as assistance with the

preparation of business cases when users want to acquire a terminal of their own. IC staff should not write code for users. It is strongly believed that if they do so, they will also have to maintain the applications. In such a case, the IC will not be different from a systems group, while the IC concept calls for helping users become independent.

Data security is a major issue at A. According to a senior IC consultant, GIS was preferred to EASYTRIEVE, as a query language, for security reasons. While it is not as easy to use as EASYTRIEVE, GIS has a "built-in file security system and EASYTRIEVE does not." In order to better ensure data security, users have access to copies of production files, not to the files themselves. Users are also advised not to develop production systems.

3.3 Success of UDA

According to the manager of Operations and Systems Development, success of DP's effort to support UDA, resides in DP's ability to demonstrate to top management that users do not waste resources, but most importantly, that the use users make of the computer is profitable to the company. The manager explained the strong emphasis put by top management on the profitability of the IC by their concern with the very rapid growth of IC usage. This growth implied increasing costs, but "proof of revenues" was not provided. This concern explains the mandate of the new IC manager, which is to control usage.

The Operations and Systems Development manager stated a second measure of success for the IC; that is, "to keep the users, and to keep them happy." It was felt that the rapid growth of usage did not leave the IC with enough time and resources to be devoted to keeping the current users happy. Consequently, improving user satisfaction is the second part of the mandate of the new IC manager.

The decrease in the backlog and in the maintenance load is not, according to the Operations and Systems Development manager, a measure of success for UDA at A. As he put it: "The backlog does not change. Why? Because there is such a large amount of latent requests that when users feel there is some slack (or a slight decrease in the backlog), they will put requests for systems they would have previously waited for." DP has not made attempts to measure the decrease in the maintenance load which could be due to UDA.

3.4 Concerns, Issues, and Problems

According to the Operations and Systems Development manager, many of the current concerns with regard to UDA at A are closely related to the dramatic growth in usage, and to the difficulties of controlling this growth. An important concern is related not only to the rate of growth but to its direction. While it was believed, when the IC was implemented, that users would develop "one-shot, small applications," some users are now developing "full-fledged applications, where data is entered, files created, updated and reports produced." However, since no editing is performed when data are input, and that there is no audit trail, DP is concerned with the risks that some of the applications

users develop provide erroneous information. However, it seems that users do not share this concern. The IC manager reports that several users have said: "If I was going to do the same thing by hand, I might make the same errors, and I would assume the responsibility of these errors." While appreciating the power of this argument, the IC manager adds that many people in the user community are more confident in the accuracy of information when it is printed on a computer output: "While one might verify the calculations when the results are typed, many people put more faith in a computer output." A does not have a ready-to-use answer to this concern. It is believed that users should be given guidelines on how to use languages, how to test programs, and so on. In addition to it, while it is effectively impossible to audit all the applications developed by users, the auditing of some applications, as well as user awareness that an auditing process exists, might motivate careful verification.

Another DP concern with regard to UDA is data duplication. It is believed that some wastage of resources results from data duplication. Both the IC manager and the Operations and Systems Development manager strongly feel that a data dictionary would help to alleviate this problem.

User friendliness of the tools, or more precisely, the lack of user friendliness of the tools is an issue for DP at A. According to the manager of Operations and System Development, many people are intimidated by the computer, and the tools which are available now do not contribute to make users feel more confident. The same manager

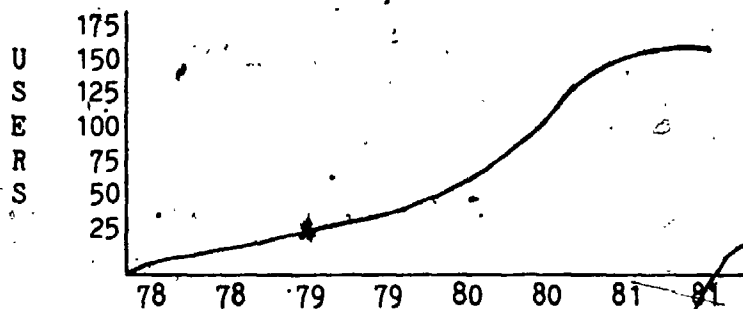
comments that some users feel that the IC consultants do not have enough expertise with the programming tools. While agreeing with it, the manager argues that in order to become an expert with the tools one has to program a lot, and on a regular basis. Such is not the role of the IC staff. However, some users spend much of their time developing applications and using the software tools. "Not surprisingly, some users may become more proficient than the IC staff, and the latter have difficulties answering their questions."

A final issue with respect to UDA is related both to the fact that some users develop production systems and to the relatively low priority that IC users have with computer operations. "When the computer goes down, users of the IC go at the end of the queue," comments the manager of Operations and Systems Development. This creates concerns for users in general, but more serious one for those managers who receive periodical reports from applications developed in the context of UDA.

3.5 The Future of UDA

Figure A.3 illustrates the change in the growth rate of the number of users since the new IC manager has been into office. The shape of the curve has been suggested by the Operations and Systems Development manager with the purpose of illustrating the change in the growth rate, rather than being an exact replica of the actual figures.

FIGURE A.3
GROWTH OF THE IC USAGE



In the near future, users will be charged for their use of the IC resources (CPU time plus IC overhead). The chargeout is seen as a means of making users aware of the costs they incur when they develop applications.

The Operations and Systems Development manager indicates that the IC might eventually move in a user area which would have developed some expertise with the use of the tools.

FOOTNOTES.

[1] It should be noted that APL is not "supported" by the IC, which means that users "are on their own." APL training is an on-line course.

COMPANY B

1. The Company

B is an insurance company with assets in excess of \$4 billion and with annual sales of new insurances policies of more than \$4 billion. The company employs approximately 4,500 people, of which 1,500 are located at the head office. B has some 130 regional offices with more than 5,000 sales representatives and 800 staff employees.

For the coming decade B management has set its goal as being a leader in the insurance industry. Management has identified four critical areas which have to be dealt with in order to meet the corporate goal. Those areas are: (1) sales, (2) productivity, (3) profits, and (4) customers services.

2. DP Profile2.1 Basic Facts

B has been using computers for more than 20 years. For the coming years, DP has received the mandate to participate to the achievement of the leadership goal of the company. According to the Office Systems manager at B, "DP management was given the challenge to demonstrate how technological developments can be exploited to help achieve the company's objectives."

The main features of the technical environment of DP at B are an IBM 3081, 13 communication lines across Canada, and more than 500

display terminals. B uses the IMS data base-management system.

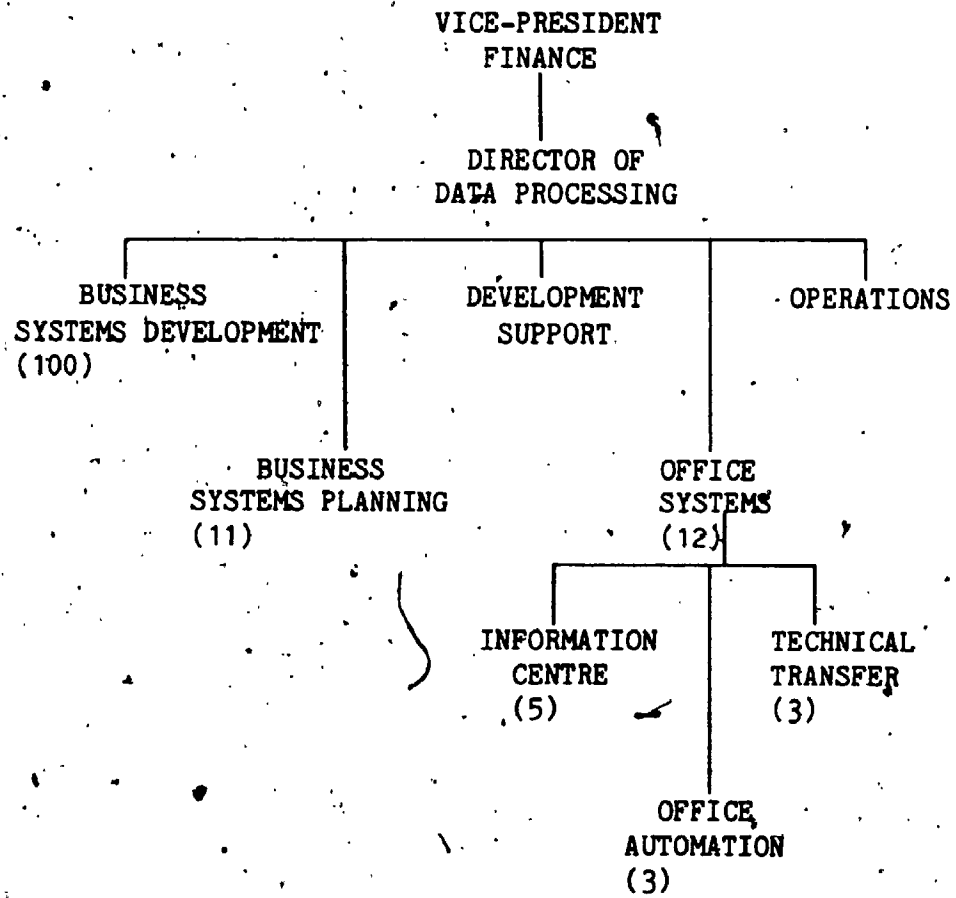
2.2 DP Organization

The Director of DP reports to the Vice-president Finance of B (see Figure B.1). DP employs 160 people who work in five major groups: Business Systems Development, Business Systems Planning, Development Support, Operations, and Office Systems.

The Business Systems Planning group is responsible for ensuring the direction and evolution of DP in the future. The role of the group is to determine, given the business B is in, which systems will be required in the future to support the mission of the firm. The group also participates in the elaboration of policies for DP. For instance, the group is currently conducting a study on Distributed Data Processing (DDP), in order to suggest a corporate policy about DDP.

The Development Support group has two major roles. First, the group is responsible for Data Administration. According to the Director of DP, the role of DA is "to ensure a corporate perspective on data." The group spent 18 months cataloguing data elements in the existing systems and has identified approximately 8000 of those elements. The group is also responsible for ensuring the data security, and works in close relationship with the Information Centre (IC) and with Business Systems Development. The second major role of the Development Support group is to ensure common usage, within the Business Systems Development group, of systems development and programming standards. The Business Development group is responsible

FIGURE B.1
DP ORGANIZATION AT B



for actually developing the Business Systems, while Operations is responsible for running these systems.

Office Systems is a group of 11 people, plus one manager, who are responsible for researching, experimenting, and implementing the concept of Office Automation. In the future years, B wants to achieve a situation where all the office work of the office worker will be performed through a single work station. Evolving toward Office Automation is one way to participate to the increase of productivity,

which is very important for B management nowadays. Office work includes tasks such as data extraction, calculations, report preparation, personal computing, communications with other office workers, and so on. At B, UDA is seen as being part of the Office Automation philosophy and is a first step in that direction. As such, the IC is part of the Office Systems group.

2.3 Relationships with the User Community

At B, a DP Steering Committee is in place. Members of the committee are the company's Vice-presidents and the Director of DP. Users are charged for their use of DP resources. In the present context cost-justification of projects is of major importance in assigning priorities for development by DP professionals.

According to the Office Systems manager, given the corporate goal of industry leadership, "it was realized that managers of the various functions had to exploit the power of the computer to a much greater degree than they previously had done to be effective." Such a movement toward the increased use of the computer to replace more traditional methods was noticed during user interviews.

3. UDA at B

3.1 Background.

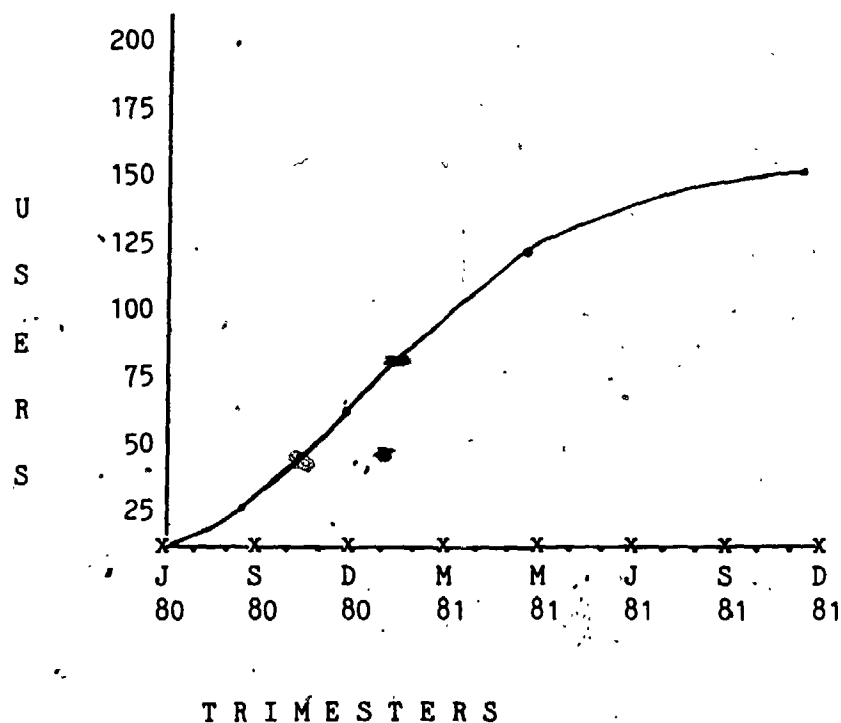
At B, the IC was implemented in the middle of 1980. As mentioned earlier, the implementation of an IC was a major step in a movement toward Office Automation, which in turn is aimed at improving the office worker productivity. However, according to the DP Director at

B, other motivating factors existed as well. These factors are the following.

As of 1979, there were several requests for new applications which DP could not fill. First, the application backlog was very large, and second several applications were not of the type for which DP professionals were trained. Frustration developed, both from the part of users, and of DP professionals. The former were unhappy, because they knew data were "somewhere in the system" but they could not get access to them. DP professionals were also upset because they had to work on applications they did not like. As put by the Director of DP, having users develop some of their own applications would "free us [DP] to do the things we do best."

In late 1979, a cost-benefit analysis was prepared in order to support the idea of implementing an IC. Two major categories of UDA-related benefits were identified, namely, productivity improvements and value of improved decision-making capabilities. Productivity improvements result from (1) computerized applications that "were done manually or by another method previously," or from (2) "applications that have to be done to meet government or external regulations and requirements or to meet top management requirements" [1]. Value of improved decision-making results from better information provided by an application or from the consideration of more alternatives in cases where an application permits to address "what if" types of questions.

FIGURE B.2
GROWTH IN NUMBER OF IC USERS



The IC began its operations in June 1980. IC management expected to have 15 users by the end of the sixth month of operations. Actually, they had 60. Figure B.2 illustrates the growth of the IC in terms of number of users, over a 21 month period. It is estimated that approximately 500 of the 1500 head office employees are potential IC users. One limitation to the growth of usage of the IC is computer capacity. Computing facilities are upgraded on a regular basis, However, the demand for usage grows faster than the computing facilities are upgraded.

3.2 UDA Now

The IC is staffed with five consultants who report to the Office Systems manager. Six terminals are available to those users who do not have terminals in their own department.

The IC has the mission of helping users help themselves and making them independent. The task of the IC consultants consists of training users, providing debugging assistance, assisting users in benefits tracking, meeting with user departments in order to assess "how things are going," and promoting the IC services to user departments. IC consultants are also responsible for upgrading their teaching material and keeping up-to-date with the programming tools they support.

The following tools are made available to users through the IC. GIS and APL/DI are query languages. The IC advises users to use APL/DI when the files they query are not too large; use of GIS is more advisable otherwise. ADRS-II is a reporting tool, while GDDM and SAS/GRAPH are the graphics packages. Finally, APL is provided as a personal computing tool, while DCF is used for generating text.

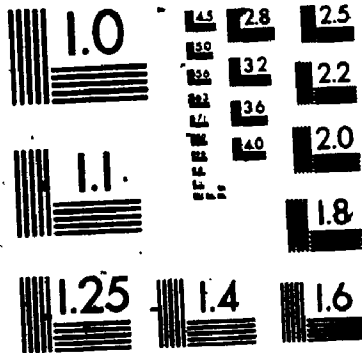
Training takes place on a one-on-one basis. Training includes learning how to use programming tools but also provides users with the standards they should use. Users are charged for the training they receive, as well as for CPU time, terminal connect time, input/output time, and lines printed.

As a rule, IC consultants should not write code for users. However, following a top management directive, the IC staff is responsible for writing code for users in the context of special (urgent) applications. According to the Office Systems manager, such actual development of applications is something from which the IC should stay away. First, it requires time from consultants who cannot perform their other duties as well. Second, problems arise once the IC has developed an application and has handed it back to users. Typically, users who do not know the mechanics of the application are reluctant to make changes when necessary, and do not seem to be as satisfied with the services received as users who develop their own applications.

3.3 Success of UDA

The IC was introduced more as a means of increasing user productivity and the company's profitability rather than for direct benefits to DP. According to the Director of DP, the decrease in the backlog and in the maintenance load was not considered as a benefit to be obtained from UDA, at the time the IC was implemented. Since the introduction of the IC, the backlog has not reduced. Given the decrease of hardware costs and the increase of manpower costs, new systems that were not cost-effective in the past become cost effective, and user departments request their development. As put by the Director of DP, "if we assume that users would have asked us to develop applications they are developing themselves, then UDA might have an effect on the backlog. However, it is almost impossible to determine it."

5



The primary measure of success of UDA at B is defined as the profitability for users in particular and to the company in general, of the applications users develop. User satisfaction with the IC services is another measure of success.

For DP, measurement of the benefits users get from UDA is critical because users use an increasing proportion of the company's computing resources, and DP shares with users the responsibility of justifying levels of use. In order to do so, the manager of Office Systems puts efforts into tracking UDA benefits. For instance, in April 1981, users were asked to evaluate the benefits they obtained from the applications they had developed. A document entitled "Guidelines for User Business Cases" assists users in the evaluation process. The document outlines the following potential benefits of applications users develop. These potential benefits are:

1. Productivity improvements
2. Cost reduction/cost avoidance
3. Profit increase
4. Earlier achievement of benefits
5. Lost benefits due to missed deadlines (without the application)
6. Value of improved decision-making capabilities
7. Headcount avoidance
8. Headcount reduction
9. Intangible benefits

For DP charging users for the services received by the IC is a way of ensuring that the resources are used in a way that is profitable to the firm.

3.4 Concerns, Issues, and Problems

Accuracy of the information provided by the applications users develop is a concern for the Office Systems managers. While the IC has the prerogative of auditing user programs, the number of applications developed by users is too large to make it possible in each case. For instance, there are more than 600 GIS programs which have been developed by users. However, if a job appears to take an abnormal length of time to run, it will be verified and users advised to make required changes. In order to further assure quality of information, emphasis is put on using programming standards at the time users are trained with programming tools. According to the Office Systems manager at B, auditing some user application and emphasizing the importance of testing and of the use of standards are the main tools which are available to DP to assure quality control of the applications while complying to the philosophy of "user independence" underlying the concept of an IC.

The fact that some users develop applications wherein files are created and updated is not a major concern to the Director of DP at A. As this manager puts it: "There are data that belong to the corporation. We identified those data and control them. There are other data in which, from a corporate point of view, DP is not interested."

Another issue related to UDA at B pertains to the rapid growth in the use of the IC services. The IC has to face an increasing demand for its services. However, the growth of the IC is limited by the available computer resources. Because of this limited capacity, DP has recently approved the acquisition of a microcomputer by a user

department. This department immediately required additional computer capacity for a special stand alone project. Since the mainframe was not going to be upgraded in time to meet the requirement, DP approved the acquisition of the microcomputer.

In the opinion of the Office Systems manager, there is no real problem with microcomputer being used by user departments. However, DP wants to ensure that if needed, those computers can be connected to the mainframe. The Technical Transfer group, of which the Office Systems manager is responsible, is currently evaluating microcomputers in order to advise on a policy about the acquisition of microcomputers.

According to the Office Systems manager the relatively low degree of user friendliness of the tools available to users is another UDA issue. This manager indicates that at the present time, most users are more of the professional or clerical type than managers. The manager suggests that one reason for that is that the tools available to users still require a fair amount of training and practice before one can become proficient with their use.

3.5 The Future of UDA

At B, usage of the IC is expected to continue to grow. In the future DP expects to develop and instal new tools and services while ensuring the quality of service provided to the user community by the IC.

According to the Office Systems manager, the implementation of the IC concept was a critical component of the movement toward Office Automation. In the relatively near future, DP expects to implement, on

a broad scale, the following services: word processing, electronic document retrieval and distribution, computer-aided instruction, video conferencing, as well as personal computers.

FOOTNOTES

[1] From internal document entitled "Information Centre Guidelines for Users Business Cases."

COMPANY C

1. The Company

C is a utility with assets of approximately \$1.3 billion and sales of \$1.1 billion. Eight hundred people work at the corporate head office and the company has 9 regional offices, for a total of 2700 employees. The largest proportion of the company's activities takes place in Ontario but it also has operations in Quebec and in New York state. C is a wholly owned subsidiary of a parent organization.

2. DP Profile

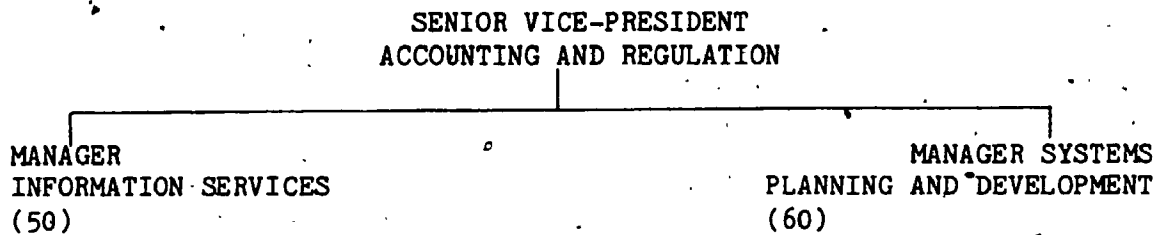
2.1 Basic Facts

C has been using computers for commercial DP for approximately 20 years. DP employs 128 people, in two major groups: Systems Planning & Development and Information Services. Computer power is provided by an IBM 3031 on which development and testing are performed and an AMDAHL V6 which is devoted to production systems. There are approximately 400 terminals installed in the user community. These terminals are mainly used for on-line inquiries of production files. Some of them are used in the context of UDA. The monthly rental-equivalent for hardware and communication costs is approximately \$225,000.

2.2 DP Organization

Two managers, who report to the Vice-president Accounting and Regulation are in charge of DP activities: the Manager of Systems Planning, & Development and the Manager Information Services (see Figure C.1).

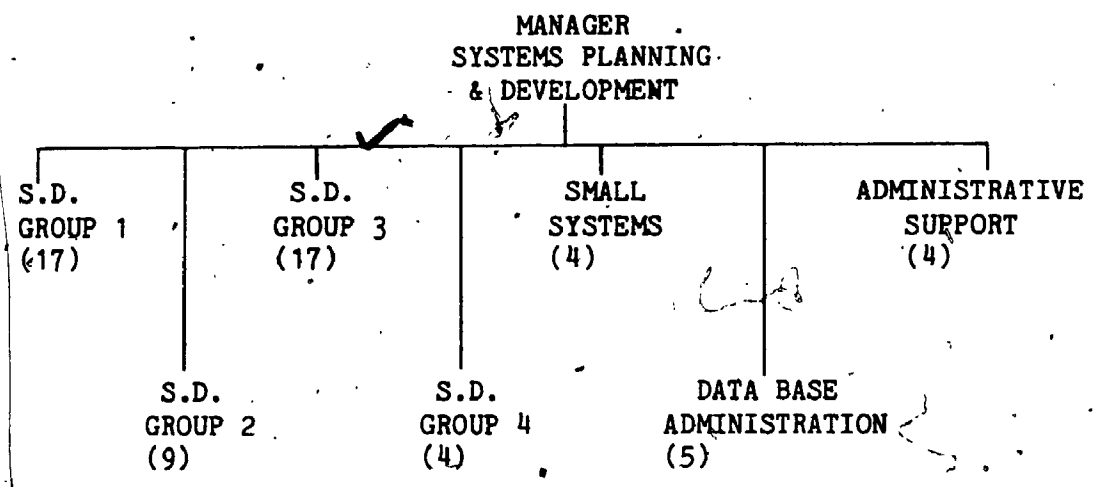
FIGURE C.1
DP ORGANIZATION



Information Services is responsible for computer operations, production control and systems programming. Systems Planning and Development is divided into seven groups (Figure C.2). Data Base Administration is responsible for data base design and maintenance. The Data Base Management System used is ADABAS. Four groups are responsible for systems development per se, with regard to the following functional areas. A first group is in charge of human resources, operations, engineering, accounting and budget, economics and statistics systems. Another group has the responsibility of materials management and marketing functional areas and a third group deals with customer accounting systems. The fourth development group is in charge of customer billing systems and also has the

FIGURE C.2

SYSTEMS PLANNING & DEVELOPMENT ORGANIZATION



Note: S.D. is used for Systems Development
 : EASYTRIEVE support provided by one person in Group 4
 : MODELER support provided by Small Systems

responsibility of training coordination, security, and user support. Within this group one person is responsible for supporting EASYTRIEVE which is one of the programming tools used in the context of UDA. The Small Systems group is responsible for responding to sudden, urgent requests for applications, of the scientific or decision support type, as well as for prototyping. In addition to these responsibilities, the group provides support to MODELER users [1]. Finally, a group of four people provides administrative support to the department.

2.3 DP from Inside

Approximately 75 percent of the DP budget is spent on developing,

operating, and maintaining operational control type of systems, 15 percent for management control systems, the remainder being spent on systems supporting strategic planning systems. DP uses formal project development and project management methodologies, and standards exist for programming documentation, and so on.

2.4 Relationships with the User Community

The Senior Executive Group, composed of the President of the company and of four Vice-presidents is also the DP steering committee. The group meets twice a year during which meetings DP presents future projects and products. Unless there is a major "resource problem" the major role of the committee is to approve the budgets. Once a year, DP management meets with middle management of the user community in order to determine which systems users want DP to develop. These meetings take place in order to set priorities for systems development; as put by the Systems Planning and Development manager, those meetings are "smoothing operations," and final agreement is reached on a negotiation basis. The main criterion DP uses to set priorities is the benefits a system will bring to the company; a good business case is critical for a new system to be developed. To date DP does not have a large backlog, and, according to the manager of Systems Planning and Development, it is "quite the contrary." As an explanation for the small size of the backlog this manager suggests the following. First, most of the core systems of the company are already computerized. Second, users seem to perceive DP services as being too expensive. This part of the explanation is related to users being charged for their use of DP resources and to the company's rigid budgetary

controls. Users are directly responsible for the rental charges of their terminals and DP charges them for terminal usage, data entry, CPU time, production control, etc, and the Systems Planning and Development budget is cleared to the users. Those charges are part of users' budgets. During the past years, the company's budgetary controls have been tightened. Consequently users have tried to keep their costs low and cutting DP costs or avoiding new DP costs seems to have been one way of doing so.

No formal DP groups exist in user departments. However, most departments have either one person or a group of people who are intermediaries between their department and DP. The main responsibility of such groups is to support their department in their use of production systems, and to monitor the changes required by users.

Outside timesharing usage is relatively low at C. It accounts for approximately \$3000 per month, and consists mainly of use of special packages for financial analysis and of use of public data bases by librarians. There are a few (two or three) micro computers installed in user departments. Recently, the Senior Executive Group set the policy that DP ought to be involved in the acquisition of any "computer type" of equipment.

3. UDA at C

Two major tools are available to users at C for application development: MODELER and EASYTRIEVE. The former is an in-house developed modelling package. Both tools have been in place for several years; EASYTRIEVE has been used for more than six years, while the development of MODELER started in 1973. Support of the tools is provided by two separate groups within DP: MODELER is supported by the Small Systems group, while one consultant from one of the Systems Development groups provides EASYTRIEVE support. The manager of System Planning and Development estimates that less than 10 percent of the total CPU hours used in the company are used in the context of UDA. Three pools of terminals are available for those users who do not have terminals in their departments.

3.1 MODELER

MODELER is a model-building package which was, (and still is) developed by the current manager of the Small Systems group. In 1973, he was assigned to the task of computerizing the company's manual financial model. As the manager puts it:

It was a straightforward piece of programming, resulting after two or three months in a fixed logic FORTRAN program which adequately ran a five year forecast, in a model of about 500 items. The need for flexibility became apparent very soon, and the problem of documenting the model and changing its logic loomed ominously. Prodigious efforts were required whenever the logic (or even the reports) of the model had to be changed -this was happening with alarming frequency- and I couldn't see myself ever writing comprehensive documentation....I set to work on some powerful extensions to the program which eventually allowed the

model-user to design his own reports and specify his own logic; at this point the program was also able to produce its own documentation. [2]

Since that period, MODELER has continued to evolve, its designer adding new features. Efforts were also made to spread the use of the modeling package. The manager of Small Systems recalls that:

In the early days I used to go around looking for people pushing pencils around rows and columns: I would ask a few questions about what they were doing, and then, if it seemed appropriate, explain about MODELER. The user of the first two or three models, who was then the Budget Manager, helped me to persuade these people on MODELER....Later, colleagues of MODELER users, curious about all the computer printout, would soon learn that they could use MODELER if they wished, and people started coming to me to find out how they could get started. [3]

During the first years of existence of MODELER, new users "would materialize in ones or twos," and training took place on a one-on-one basis.

Nowadays there are approximately 50 MODELER users and 150 models.

The models vary in size and in complexity:

At one extreme there are models which are run monthly to produce a single-page report, at the other extreme, the financial model is often run many times a day, and produces 50 reports per [subsidiary] company....Some models calculate most of their items, others receive most as input from other models and are used as little more than report generators. Some are run by and for office and clerical staff, others go directly to vice-presidents or...our regulatory authority.

[4]

Training is now provided on a more formal basis than during the first years of MODELER's existence: classroom training is combined with workshops at the terminals. Assistance with design and programming of the models is also provided to users.

3.2 EASYTRIEVE

Before being made available to users, EASYTRIEVE was used by Systems Planning and Development people under the support of the Data Base Administration group. At that time, approximately six years ago, the DP backlog was quite large. Making EASYTRIEVE available to users was seen as a means of off-loading DP. Responsibility of EASYTRIEVE support was given to one of the systems development group.

Nowadays, one consultant is in charge of providing support to approximately 50 EASYTRIEVE users. Several EASYTRIEVE users are MODELER users as well. Training with EASYTRIEVE consists of an audio-visual course, and an "hands-on session" following the course. In addition to the responsibility for user support the consultant also develops EASYTRIEVE applications for user departments.

3.3 Control of UDA

On the topic of how the use of MODELER is controlled, that is, how DP "chooses to allow whom to do what with a model," the manager of the Small Systems group states that:

No control is extended in this way, just as no-one ever attempted to control the use of 15-column spread-sheets. Within departments the normal supervision process determines

that no-one is using MODELER frivolously. But across departments there is no guarantee that several people are not calculating "property, plant and equipment" by different methods in different models (and probably getting different answers) -but again- there never was. You will generally find that each party has valid reasons for calculating the figure its own way, and distrusts the number provided by another department. [5]

EASYTRIEVE users have read only access to the actual production systems files. DP has the responsibility of making any changes to DP written code. The Systems Planning and Development managers says that, even if users have not been told explicitly, they should not develop "major production systems." The same manager thinks that DP should not be concerned with the correctness of the information provided by the applications users develop themselves. DP provides users with training, and makes support available to them. It is the ultimate responsibility of the users to make sure that the information they get is correct.

3.4 Success of UDA

According to the manager of Systems Planning and Development, "UDA is successful from a DP point of view "if you don't hear from the users." That is, if users are satisfied with the service provided to them, then UDA is successful.

To the manager of the Small Systems group, success lies in the increased usage of MODELER and in user satisfaction. In his opinion, the following are important reasons for explaining the actual success of MODELER. First, the package was developed "in the real world of

serious users," the designer keeping a close contact with the users and being "in a position to react quickly to problems and suggestions." The availability of consultants is another important factor of success, as well as seminars and newsletters "which encourage the expression of the new ideas, and serve to keep users up to date regarding both what MODELER can now do and is now doing" [6]. A readily accessible computer system, and terminals at users' proximity (i.e. in their own departments) are also considered as being related to the success of the modeling language.

3.5 Concerns, Issues, and Problems

To the manager of Systems Planning and Development, the main issues with regard to UDA are related to future developments. In a relatively near future, instead of being provided by two different groups, user support will be under the responsibility of a single group, which is likely to be the Small Systems group. A broader variety of tools will be made available to users. Consequently, the responsibilities of the support group will be different from what they now are. According to the Systems Planning and Development manager, the new UDA context will require from the support group, the ability to evaluate new programming tools. It is also necessary that the consultants know well the company's production systems so as to be able to assist users in finding the data they need and better understand their problems. A first issue is to make sure that the support staff has the required skills, and is interested in that type of work.

The main concern for the Systems Planning and Development manager is "one of control." UDA is a very rapidly growing area and vendors are pushing their products quite aggressively. What is important is "to have the right product, at the right time, used by the right people, in the right way." This implies that DP has to know user needs, and be able to evaluate products. From the user side, people who will understand the limitations of the tools should be those chosen for receiving training. The latter is related to the experience DP had had with some users being trained with EASYTRIEVE and not being able to use it to their own satisfaction or to the satisfaction of their superiors. According to the Systems Planning and Development manager, while quality of training and easiness of use of the tool itself have to be taken into account to explain the above, user ability (or skills) to use programming tools also have to be taken into account.

3.6 The Future of UDA

C is evolving toward a situation wherein the support of UDA will be provided by a single group, in an Information Centre type of environment, and where a broader variety of programming tools will be made available to the user community. One of the reasons why C is going in that direction is that the demand for it exists in the user community. As put by the Systems Planning and Development manager, "if DP does not do it, users will do it themselves." The manager believes that UDA will be more successful if it exists in a "controlled environment." This manager does not think that more widely spread UDA will contribute to further decrease the backlog; "it will likely do the opposite." He explains the potential of UDA to increase the

backlog by the fact that most programming tools currently available to users have limitations. UDA creates new ideas for application development for users but either the tools or the users themselves have limitations. In such a case, users are likely to come to DP to put these new ideas into practice.

FOOTNOTES

- [1] MODELER is an in-house developed modeling language
- [2] Presentation made by the manager of Small Systems group and designer of MODELER
- [3] Ibid.
- [4] Ibid.
- [5] Ibid.
- [6] Ibid.

COMPANY D

1. The Company

D is a manufacturing company, with assets of more than \$2.5 billion and sales in excess of \$8 billion. D is a wholly owned subsidiary of another company; it employs 39,000 people, of which 6,000 are salaried employees, the remainder being hourly-paid workers.

2. DP Profile2.1 Basic Facts

At D, the DP department is responsible for the development and operations of all the computer-based applications of the firm in Canada. It is a policy of the parent company that user departments should use host computer of their own division, when the division can satisfy their requirements. If not, user departments can use the corporate timesharing services. If those services still do not satisfy their requirements, users can go to outside timesharing firms or purchase their own computer. The acquisition of computers by user department has to be approved by DP. Presently, most of the systems are run on host computers controlled by DP. These computers are one IBM 3081 and one Amdahl V7. DP employs 300 people.

2.2 DP Organization

At D, the DP department consists of three major groups. First, Development and Maintenance is responsible for systems analysis, design, programming, and maintenance. In many instances, systems are designed by systems analysts who are based in user departments. In such cases, the Development and Maintenance group's involvement is limited to programming activities.

The Technical Support group provide services with respect to computer networks, systems programming, and data base administration. The IC is part of the Technical Support group.

Finally, the Operations group manages the computer operations, and is in charge of data entry, and of the data centre activities.

At D, several departments have their own systems groups; presently, eight of the major departments have such groups; for instance, there exists a Material Management Systems Group, Customer Services Systems Group, Payroll and Benefits Systems Group, Manufacturing Systems, Personnel Systems Group, and so on. Of the nine major plants, some are starting to develop their own systems groups. The main responsibility of these systems groups is to analyze and design systems for the department they are in. Programming is the responsibility of DP. Many of the systems analysts in these groups were transferred from DP; for those who are hired directly by user departments, a training period spent in DP is most common. Systems

groups are also responsible for transmitting requests for systems maintenance, from users to DP. Most of these analysts also develop "data extraction" applications for users. For several years, the tool used for doing so was EXTRACTO, whereas now it is GIS.

2.3 DP from Inside

The largest proportion of the DP budget at D. is devoted to the development, operation and maintenance of operational control type of applications. Fifteen percent of the budget is spent on management control applications, the remainder being spent on planning type of applications. DP uses a systems development methodology, namely, PRIDE. Standards for programming are also into place. The Data Base Administration (DBA) group is involved mainly in data base design as well as in data base maintenance. The data base management system used is IMS, and there is an on-line data dictionary.

3. UDA at D.

3.1 Background

In June of 1979 an Information Centre (IC) was implemented at D. According to the Director of DP, the IC was implemented in order to increase DP responsiveness to user needs. There existed several requests for small one shot applications, that DP could not fulfill. The IC is part of the (DBA) group mainly because the IC users heavily interface with the data base and DBA can more easily provide support to the IC. The IC received the mandate of providing users with capabilities and access to the company files with the necessary

security and accounting controls.

Before the implementation of the IC, one tool was available to analysts in user systems groups to perform data extraction, namely, EXTRACTO. There existed some problems with the use of EXTRACTO. First, the analysts had to write the queries and then have them punched, send them to DP where they had to be checked by a programmer, and the job prioritized. The turnaround was not good and analysts were dissatisfied. GIS was the first IC tool to be introduced. As put by the IC Supervisor, when GIS came in, "classes were filled in advance." Most of the early GIS users were systems analysts from user systems group. Even nowadays, GIS users are still dominantly systems analysts: GIS is seen by most people as being a replacement of EXTRACTO, and systems analysts have traditionally been responsible for developing this type of applications. Moreover, some user systems groups are reluctant to let end users develop applications.

ADRS-II, PLANCODE, and APL were also introduced. The usage of these tools did not spread as much as it did for GIS. However, in the first few months of the IC operations, ADRS-II, APL and PLANCODE were not available to those users who were located in other buildings than where DP was located [1]. When they did become available to other locations, usage started to grow rapidly.

According to the IC Supervisor, users of ADRS-II and PLANCODE are different from GIS users. While GIS users are, in general, DP analysts who work in user areas, ADRS-II and PLANCODE users tend to be non-DP

professionals. Training of such users require more emphasis on explaining the workings of a computer, as well as familiarizing users with the use of programming tools..

The total number of trained users for the IC grew quite rapidly from 1980 to 1981: at the end of 1980, there were 69 trained users while there were 250 of them at the end of 1981. In 1980, a study was conducted, and evaluated the potential number of users as being around 1200 people. However, according to the IC Supervisor, all trained users do not remain active. Table D.1 compares active vs. trained users for each of GIS, PLANCODE, and ADRS-II, as of January 1982 (an active user has made use of the computer during the past three months) [2].

TABLE D.1

IC USERS: TRAINED VS ACTIVE

<u>TOOL</u>	<u>TRAINED</u>	<u>ACTIVE</u>	<u>% OF RETENTION</u>
GIS	157	127	81%
ADRS-II	54	34	63%
PLANCODE	38	27	71%
	<hr/>	<hr/>	<hr/>
TOTAL	249	188	76%

- According to the IC Supervisor, the retention rate was lower in the past than it is now. He further explains the (relatively) low retention rate by the fact that some department supervisors asked to

have several of their people trained at the same time, yet, not having applications to be developed by all those people. Nowadays, in order to increase the retention rate, the IC asks that users have applications in mind before receiving training.

The IC staff prepared a document entitled "INFORMATION CENTRE User Handbook"; the 35 page document has been available to the user community since December 1980. The document introduces the IC concept, provides "product selection guidelines," outlines the IC services, etc. For each programming tool (i.e. GIS, ADRS-II, and PLANCODE) the brochure briefly describes the following: language overview, recommended usage, non-recommended usage, resources required, skill requirements, education requirements, rules for file access, and restrictions.

3.2 UDA Now

Nowadays, the IC, with three consultants and one supervisor, supports 190 users. The IC has six terminals, two of which are reserved for the IC staff, the remainder being used mainly for training purposes. Not many users actually come to the IC, either to work there or to get assistance. The main reason is the physical dispersion of the user community. Several users are located in cities other than the one where the IC is. Others, even if they are in the same city and in the same complex of buildings would have a few to several miles to do in order to get to the IC. The IC strongly encourages users to have a terminal available near their work station before receiving training. However, according to the IC Supervisor, most users have access to

terminals in their own areas; he estimates that more than 1100 terminals are installed throughout the company.

The tasks of the IC staff include training, debugging assistance, as well as technical and administrative support. GIS training is a four-day course, offered approximately once a month. PLANCODE training is also a four-day course while ADRS-II is a one-on-one, half a day course. DP education group offers a four-hour "video and hands-on" course to familiarize users with the use of terminals. This course is a prerequisite for the courses offered by the IC (if the user does not have experience with the use of terminals). APL training is a computer assisted course. One IC consultant is responsible for each one of ADRS-II, PLANCODE, and GIS.

Users are charged for their use of computer resources. Until the beginning of 1982, the charges were broken down under two categories: CPU time and Read-Write operations. Starting from 1982, a third category is added, namely, the number of lines printed. According to the IC Supervisor, doing so is more equitable for those users who use mainly cycles and not much printing. In addition to it, the parent company's timesharing services breaks down the charges in 8 to 10 different categories. The IC Supervisor argues that in order to remain competitive, the IC has to make efforts to break down its charges as well.

3.3 Success of UDA

According to the Director of DP, UDA, instead of participating to

a decrease in the backlog, is more likely to increase user demands. As he puts it: "It will increase the number of mainframe systems necessary to provide users with data they need." For this manager, success of UDA appears to be determined more by user satisfaction with the IC services than an increase in DP productivity.

To the IC Supervisor, success is closely related to ability to demonstrate to top management of the firm that the service offered to users, as well as the use user departments make of the computer, is profitable to the firm. This demonstration would then contribute to help the IC getting more resources to provide better services to users.

3.4 Concerns, Issues, and Problems

To the Director of DP, a current issue is the limited computer capacity. This limitation in capacity forces to slow down the growth of the IC, by stopping to "sell it." According to the IC Supervisor, the main issue he has to face nowadays is the limited resources made available to the IC. The Supervisor argues that additional resources are required, and that in order to obtain such resources, the IC has to get top management support. If the IC has DP support, it does not appear to be sufficient to get the resources necessary to offer the services which the supervisor feels should be provided. The IC Supervisor feels that it would be important to organize user meetings, wherein the IC staff could explain new features to users, and where users could share their experience. As well, the supervisor would like to see IC consultants evaluate new products, such as graphics packages. However, the three IC consultants are so busy teaching,

"troubleshooting," and performing administrative duties that no time is left for upgrading the IC services.

3.5 The Future of UDA

According to the IC supervisor, a most important objective for the IC is to consolidate and expand the services provided to the user community at D. However, in order to meet such an objective, the IC has to have sufficient resources, mostly in terms of manpower, and of computer capacity. Demonstration to top management that UDA is profitable to the company is a pre-requisite for obtaining the resources, and the IC supervisor strongly encourages users to report the cost benefits of the applications they develop.

FOOTNOTES

[1] The company has divisions located in several different cities. Moreover, even users located in the same city as the IC, and in the same building complex, are sometimes quite remote from the building where IC is.

[2] APL is not part of Table C.1 because APL training is computer assisted training. Moreover, APL usage is not widespread.

COMPANY E

1. The Company

E is a privately owned company which operates mainly in the forest industry. The company owns logging divisions, saw mills, plywood mills, pulp mills, as well as building materials distribution divisions. E also has interests in non-forest related industries. The company employs 7,000 people, 175 of whom are located at the head office. E has operations in all provinces of Canada and in the United States.

2. DP Profile2.1 Basic Facts

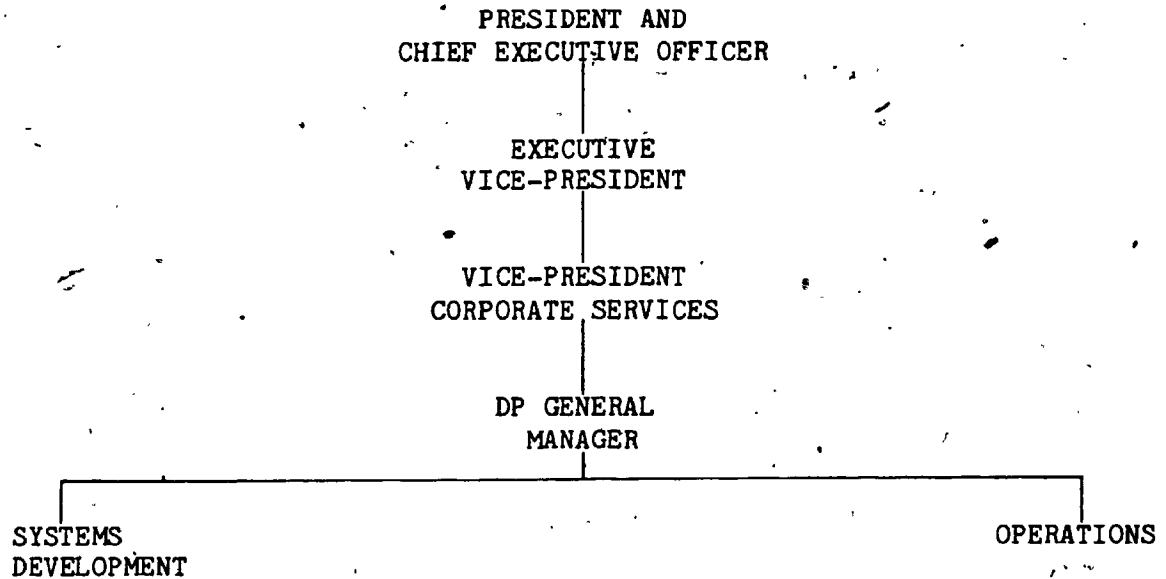
DP approximate monthly expenditures for hardware are \$32,000 and the department employs 55 people. E has been using computers for commercial DP for more than 18 years. The computer presently installed is an IBM 4341.

2.2 DP Organization

The General Manager of DP reports to the Vice-president Corporate Services, who in turn reports to the company's Executive Vice-president. Two groups constitute DP, namely, Systems Development and Systems Operations.

FIGURE E.1

DP ORGANIZATION AT E



DP is responsible for developing systems and processing data for the majority of the company's 30 divisions. Nevertheless, some divisions have their own computer. For instance, two divisions have an HP 1000; one of these divisions is very active in developing systems, while the other is not. One division has a DEC 11; this division is relatively independent, E owning only 50 percent of its shares. This division has its own DP staff. There are also plans for installing minicomputers in warehouses of the building materials division. These computers would be linked to another CPU which in turn would be linked to the head office IBM 4341. However, applications running on these computers will be developed by the head office department. The head office DP department is responsible for approving all purchase of

computing equipment. The DP General Manager is concerned with the risks related to the acquisition of mini and or micro computers by the divisions: duplication of applications, equipment not compatible, loss of control by DP of the core systems, are the main concerns.

2.3 DP from Inside

DP does not prepare long range plans. A structured methodology, with review boards is used for systems development. The methodology calls for much user involvement. Data base technology is just being implemented, with IDMS as a Data Base Management System. However, up to now a Data Administration responsibility does not exist. According to the DP manager, the backlog of applications is not large.

2.4 Relationships with the User Community

A DP Advisory Committee, constituted by representatives of the user community, reviews the requests for systems development that DP receives, and the committee establishes priorities. The committee relies heavily on cost-benefits analyses to set priorities. DP is responsible for identifying the costs of a proposed system, while the system's sponsoring division has to justify the benefits. The Vice-president Corporate Services takes advice from the Advisory Committee, but he still has the last word on what the priorities are. For the past fifteen years, users have been charged, in real dollar terms, for their use of DP services.

Some divisions have Systems Coordinators, who are located at the head office and whose role is to be a liaison between their division

and DP.

2. UDA at E

3.1 Background

Approximately 13 years ago DP developed a ledger package which users use themselves to prepare their financial statements. This package runs in batch mode and is mainly a report generator. Ten years ago, in addition to this package, DP introduced FORESIGHT. At that time, budgeting was a very time consuming task for the company's accountants. As put by the General Manager of DP: "At budget time, for a period of two months accountants would spend their time preparing budgets." DP estimated that the costs of such an exercise, in terms of time lost, was very high. Consequently they undertook an investigation into packages that would have the potential of saving time in the preparation of budgets. FORESIGHT was chosen and one individual was selected to be part of a pilot project, which showed that FORESIGHT was an appropriate tool.

MARK IV was introduced a few years ago as a tool for DP professionals; it is also available to some users, on a read-only basis, but its use is not widespread.

3.2 UDA Now

Nowadays, there is one individual, the FORESIGHT coordinator, whose full time job is to provide support to approximately 70 users. Twenty one of these users are located at the head office, the remainder

being in divisions, the location of which varies from a few miles to several thousands of miles from head office. The coordinator is responsible for training, which is provided at the division's site for remote divisions, and for providing assistance during the design and programming of applications. Yet, he does not write code for users. Presently, there are more than 6000 FORESIGHT applications. FORESIGHT users are mainly accountants, and most of them also use the ledger package DP developed. While FORESIGHT is used mainly for budget preparation it is also useful in other contexts. For instance, last year E was considering the acquisition of a company, and the decision had to be made quickly. An accountant at the head office developed a financial model and was able to provide 17 different scenarios based on questions asked by the President.

The number of of MARK IV users is low. According to the DP manager, this tool is perceived by users as being hard to use. Most of the people who use it are "fairly senior office staff."

3.3 Success of UDA

At this time, the major tool in the context of UDA is FORESIGHT. Keeping that in mind, the DP General Manager claims that UDA is successful because it contributes to increase productivity in saving large amounts of time for budget preparation.

3.4 Concerns, Issues, and Problems

Nowadays, DP does not aggressively promote UDA to the user community. DP is selective in determining who will receive training

with FORESIGHT. The main reason for doing so is that DP want to ensure that those who use the tools are competent enough to use them. Another reason for not heavily promoting UDA is that DP wants UDA to grow in a controlled fashion, as it has been the case for the growth of DP in the company. The DP General Manager states that if the appropriate tools were made available to users, he expects that the number of users developing applications would increase dramatically: "There would be a flood" and this would rapidly lead to capacity problems. The current computer system would have to be upgraded, and DP would have to demonstrate to top management that applications users develop themselves are profitable for the company as a whole. This is difficult task, because once users have the tools, DP does not really know how they use them. Closely related to the capacity issue is the fact that in the context of UDA, the demand for computer power occurs almost only during daytime. An increase in capacity to suit this demand while leaving the computer idle at night might not be welcome by top management.

On the other hand, some divisions are making strong pressures so as to become more independent from the DP department. If UDA tools were made available to these divisions, they might find that type of independence satisfactory and not ask for more decentralization.

The rapid expansion of the company and the higher demands from user, increase DP's workload. According to the General Manager of DP, "there is no sign that we will be able to develop all system we should develop." In addition to the increase in demand for services, DP faces

a shortage of DP professionals; during 1981 the department had to function with three programmers less than what was required.

In consequence, E is definitely heading toward spreading the concept of UDA, but this will likely be done in a cautious fashion.

The General Manager of DP identified two other issues related to UDA. First is the difficulty of finding user friendly languages. The majority of languages which are said to be user oriented still require long training and quite a lot of practice before one can become proficient. The second concern is related to quality assurance; that is, it is almost impossible to make sure that the information provided by applications developed by users is correct. In theory, DP could audit the applications; however, this is not actually feasible. Even now, with only one major tool (FORESIGHT), the task of auditing the 6000 applications is effectively impossible.

3.5 The Future of UDA

The DP General Manager sees the future role of DP as providing data to users and preserve data integrity. Users will have tools available to them to produce their own reports.

COMPANY F

1. The Company

F is a utility company with assets of \$1 billion and annual sales of \$800 million. The company employs 2400 people, of which 500 are in management, supervisory and professional areas, 715 are clerical staff, 255 are technical specialists, the remainder being hourly-paid employees.

2. DP Profile2.1 Basic Facts

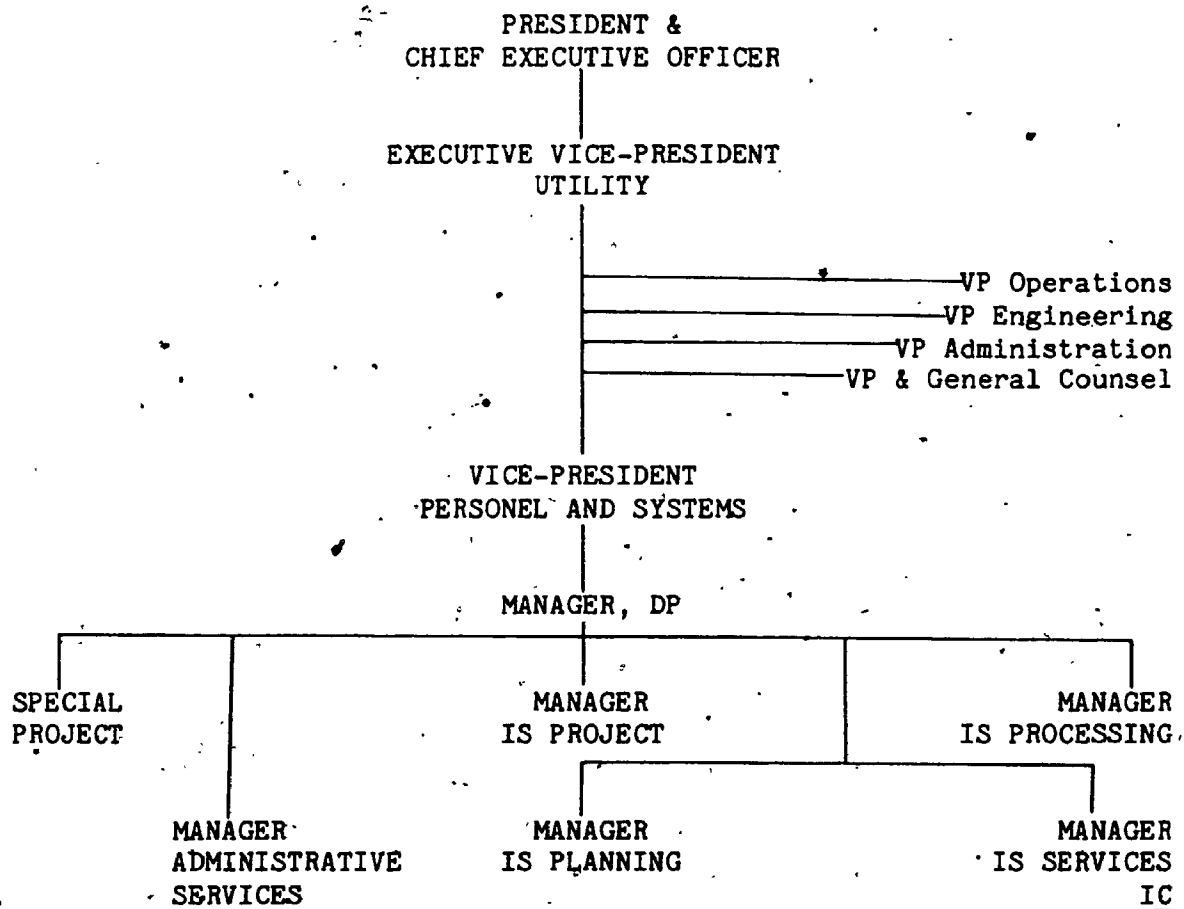
The DP department at F employs 120 people, and has a hardware and communications budget of approximately \$200,000 per month. F has been using computers for commercial purposes for more than 20 years. To date, the company has an IBM 3032 and an IBM 148; the latter is to be replaced by an IBM 4341-II. In the meantime, because of the capacity limitation of the computers, the company purchases time from an IBM Service Bureau.

2.2 DP Organization

At F, the DP activity is an independent operating department, reporting to top management. The Vice-president in charge of DP is also in charge of Personnel and reports to the Executive Vice-president Utility who in turn reports to the President and CEO. The DP department is organized in six major groups (see Figure F.1), namely,

FIGURE F.1

DP ORGANIZATION AT F



Information Systems Processing, Information Systems Services, Information Systems Development, Administrative Services, Information Systems Planning and a Special Project Group. The latter has the responsibility for revising some of the major production systems that are now obsolete.

Systems development, processing, and the management of DP are centralized. There are approximately 130 terminals installed in the company's regional offices. These terminals are used mainly for data entry and for running systems that have been developed by DP. At the head office 50 terminals are installed in user departments and are generally used in the context of UDA.

2.3 DP From Inside

In 1981, a five year plan (covering 1983 to 1987) for computing at F was prepared. The planning exercise was conducted by DP people in conjunction with IBM and with representatives of user departments at the Vice-president level. The plan calls "for installing a large-scale computer in the fiscal year 1983 to replace a smaller currently installed one, and maintaining a high level of funding for computing over the next 5 years, in order to contribute to the effective and productive growth of the Corporation" [1]. In addition to the five year plan there exists a plan for the current year, which describes in more details what has to be done by DP during the current year.

The concept of data administration has not been introduced yet, neither is there data base technology. Formal methodologies are in place for systems development (PRIDE) and for project management (PAC2). These methodologies have been in use for approximately two years. Most of the systems that are in place are operational control types of systems.

2.4 Relationships with the User Community

F does not have a steering committee. Yet, the Vice-president of the user departments are involved in setting priorities for the development of new systems. This involvement takes place as follows. Within the IS Development group, four managers are responsible for the systems of each of the following areas: Finance and Marketing, Engineering, Operations, and Special Project. For each area there exists a two year systems development plan which is prepared by each one of these area manager in conjunction with the Vice-president of each function and their senior management. The plan is revised every six months. Among the different functions, priorities are set mostly on a negotiation basis.

DP costs are treated as corporate overhead and are not allocated to users. However, user departments that have terminals are responsible for the rental charges.

3. UDA at F

3.1 Background

UDA is not entirely new at F. However, before the advent of the Information Centre (in the summer of 1981), UDA was not as widespread as it is nowadays. In the late sixties, a Personnel and Payroll system had been developed by DP. As put by a DP manager, "during the system study, it became evident that, while the standard weekly and monthly processing and reporting requirements were fulfilled by the system, much of the information needed by the Personnel people was of an ad hoc

or one-shot nature." MARK IV was purchased and one individual in the Personnel department was trained to use it in order to satisfy the ad hoc requests. At that time and for a few years after, other users did not show much interest in becoming more involved with the computer.

DP explains this little interest for UDA by giving four reasons. First, in the DP area, the demand did not exceed the supply and DP could deliver the systems in a realistic time period. Second, most users were still intimidated by the computer and they preferred to leave the responsibility of dealing with it to DP professionals. Third, the cost of hardware was still relatively high as compared to the cost of people; it was critical to use the computer as efficiently as possible. Fourth, the business environment at that time did not require that "critical information be produced, as quickly and as accurately as needed in this day and age" [2].

However, in the late seventies, both the business environment and the systems environment changed for the company. The business environment is now more uncertain and changes more rapidly. "In this type of environment, we are finding that we can no longer necessarily rely on the tried and true methods of remaining responsive to business demands. The concept of being efficient, responsive and more productive by working faster, working harder or by hiring more staff does not apply in many cases" [3].

The new DP environment is described by the Systems Services Manager as follows:

In the first place, we have at least a two year backlog of requests from the user community. As a result our ability to respond to user needs in a satisfactory manner has continued to degrade. With the degradation in responsiveness, there has been a corresponding decrease in user satisfaction. Many users do not even bother to submit requests for new systems on the assumption that the requests cannot be answered within a reasonable time frame. Those users that do submit requests would undoubtedly be much happier with a quicker response time on the part of the Systems Department.

Secondly, the availability of experienced, qualified systems people is becoming more a problem. Especially, since we are located well away from the major DP centre. At the same time, the cost of those people we are fortunate enough to obtain is going up.

Third, there is an increasing acceptance on the part of the user community, in general, of the computer as a fact of life. Many of the younger employees, in the engineering and financial areas in particular, have used the computer as a working companion during high school and university days. So these people in particular, not only feel very comfortable with computing, but see the use of computers as an integral part of their job. There is beginning to be an ever increasing demand on the part of these younger users for the power, efficiency and productivity which computing provides.

Fourth, the cost of hardware continues to decline.

Lastly, as mentioned earlier, "friendly" end-user products are quickly becoming available.

Products that will retrieve data and products that will analyse data;

Products that can be used as planning tools and products that can be used as reporting tools;

Products that can be used for the manipulation of text and products that can be used for personalized training via CRT's;

And all of these products are designed for ease of use and with the non-programmer as their prime target. [4]

The need for having users develop their own applications was then present at F. PLANCODE and APL had been available since 1978, but

still there was a need for a broader variety of tools and for deciding on how UDA would take place in the firm. A few alternatives were considered. The first alternative was to provide users with their own computers. However, as put by the Information Services Manager:

While this would certainly put computing power in the hands of the user and would eliminate the bottleneck in the Systems Department, we felt that in our particular position the loss of control would be a distinct disadvantage to the corporation. With this approach we could foresee much redundancy, wasted resource and an inability to produce cross-functional information" [5].

The second alternative considered was Distributed Data Processing, but once again it appeared to have more disadvantages than advantages. Finally, the concept of an Information Centre was seen by DP as being the most appropriate alternative for F. The concept of an IC makes possible to ensure that corporate data are taken care of, that the data remain valid and secure, and ensure that transaction processing systems are developed by people who have the skills to develop them.

3.2 UDA Now

The IC has been in place since the summer of 1981. Nowadays, it is estimated that users of the IC use approximately 6 percent of the company computing resources. The IC staff reports to the manager of Information Systems Services. It is the responsibility of Information Systems Services to provide support to the DP community and to the user community. Services to the former include support of PRIDE and of PAC2 (consulting on the use and implementation of the methodologies); Information Systems Services also provides support for MARK IV, the use

of which is restricted mainly to DP professionals, because of its ability to alter files.

The IC supports the user community. The programming tools currently available to users and supported by the IC are: ADRS-II, PLANCODE, and APL. ANSWER D/B (a query language) and GDDM are to be made available in the very near future, while FPS will become available later. The role of the IC consultants is to train users and to provide assistance during the design and programming of applications.

To date, the tools available to users are not suited for accessing corporate files. The main reason for which users do not have access to corporate files now is that DP has neither one of a security package, or copy management facilities. The present situation creates inconvenience for some users. For instance, there are users who prepare summary reports for the budgeting activity. They receive outputs from DP, and the data from these outputs have to be re-inputted in order to produce the summary reports. However, the implementation of ANSWER D/B will make corporate data available to users on a read-only basis, hence, ensuring data security.

When users first want to develop applications themselves, they usually go to the IS Development Manager of their function who will help them determine if the application they want to develop is better suited for being developed by themselves or by DP. In the former case, the IS Development Manager will give advice on the proper programming tool to use. The user is then trained and supported by the IC staff.

Because of the present capacity limitations, most of the IC users develop and run their applications using computer time purchased from an IBM service bureau. In the future, when the computing facilities

are upgraded, it is expected that they will be brought back in-house. There is no rigid limitation of the storage space allocated to users, provided that it remains "reasonable." Users are not charged for their computer time but they receive an assesment of the costs they incur (they are "memb-billed").

Several users have their own terminal or share a terminal with other users in their department. In addition, the IC has a terminal room with 5 terminals, the use of which is reserved to users.

The IC consultants are not grouped in a single area; they are "dispersed" within DP.

3.3 Success of UDA

At F, UDA is expected to bring benefits both to the user community and to DP. Benefits to the user community include both improvement of productivity and improvement in decision making outcomes. It is expected that UDA will provide management with better information because the applications are developed by people who are experts in their own field. UDA provides flexibility, permits to do more analysis, in a quicker fashion.

Having users develop their own applications should permit DP to use their skills and talents in a more efficient manner, that is, to develop large systems instead of small, one shot type of applications. DP does not actually measure the amount of resources that are devoted to maintenance, but they have a good "feel fot its importance." It is expected that UDA will contribute to reduce the perfective maintenance load. Yet, since the IC is just starting it is still difficult to

determine if it has had any effect on the maintenance load. Another benefit for DP is that the DP professionals can also use the programming tools users use, hence being more productive. Finally, for the IC consultants, new career paths are available. This is due to the fact that the consultants are exposed to what is going on in almost all departments of the company, having to deal with a large variety of users and problems.

3.4 The Future of UDA

The DP plan for 1982 called for "increase the number of IC users by 20 by December (of 1982) and an additional 20 by April, 1983" [6].

The current situation is as follows. In March 1982, eight months after its implementation, the IC had 60 users, and one month later, 30 additional people had been trained. The DP five year plan reports that there is a potential number of 350 users, including technical, clerical, professional, and managerial users.

In the coming months, the IC will put efforts into consolidating the services provided to the user community.

FOOTNOTES

[1] Excerpt from Five Year Plan for Computing at F, Fiscal 1983 to 1987.

[2], [3], [4], [5] A presentation made in 1980, by the Information Services Manager.

[6] Excerpt from "IS Department, Direction for 1982," Internal Document.

COMPANY G

1. The Company

G is a financial institution, with assets of approximately \$65 billion and employing more than 28,000 people. Most of G activities take place in Canada, but it also have world-wide activities.

2. DP Profile

2.1 Basic Facts

G has been using computers for commercial DP for more than 20 years. The approximate average monthly rental-equivalent cost for hardware and communications is \$5 million dollars. DP employs approximately 1200 people. The company has two central computer complexes, with two AMDAHL V8 devoted to production systems in one of the complexes and one AMDAHL V8, one AMDAHL V7, and one IBM 3081, on which development and testing are performed, in the other computer complex. In addition to it, there are 8 regional data centres, spread across Canada, which employ a total of 1500 people who do not report to DP.

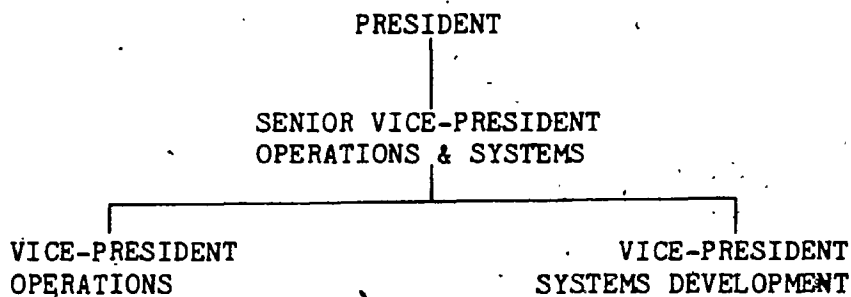
2.2 DP Organization

DP activities are under the responsibility of two Vice-presidents, namely, the Vice-president Systems Development and the Vice-president Operations. Those two managers report to the Senior Vice-president, Operations and Systems who reports to the President of the company (Figure G.1).

Systems Development. Systems Development employs 490 people, in three different locations and six groups (Figure G.2). Architecture and Design, Commercial Systems Applications, and Software & Development Services-Commercial Systems are located at one of the Data Complexes, whereas Interactive Computing Services is located at the company's main office in the same city. Corporate Systems and Software & Development Services-Corporate Systems are located at the company's head office in a different city.)

Architecture and Design is responsible for the early phases of systems development: feasibility study, study of business cases, determination of how a new system will relate to the systems' architecture already in place, and systems design. Commercial Systems Applications and Corporate Systems share the same responsibilities but with regard to different type of systems. The former deals with the systems which support the company's commercial activities whereas the latter deals with corporate systems such as personnel and payroll systems, financial systems, etc. Both groups are mainly responsible for programming the systems, according to the specifications provided by Architecture and Design. The two Software and Development Services groups have the same responsibilities with regard to different types of systems, namely Commercial Systems and Corporate Systems. Software and Development Services provide services such as ISO support and measurement of programmers' productivity. They are also responsible for systems integration and testing: the programs produced by Commercial Systems and Corporate systems are verified against the

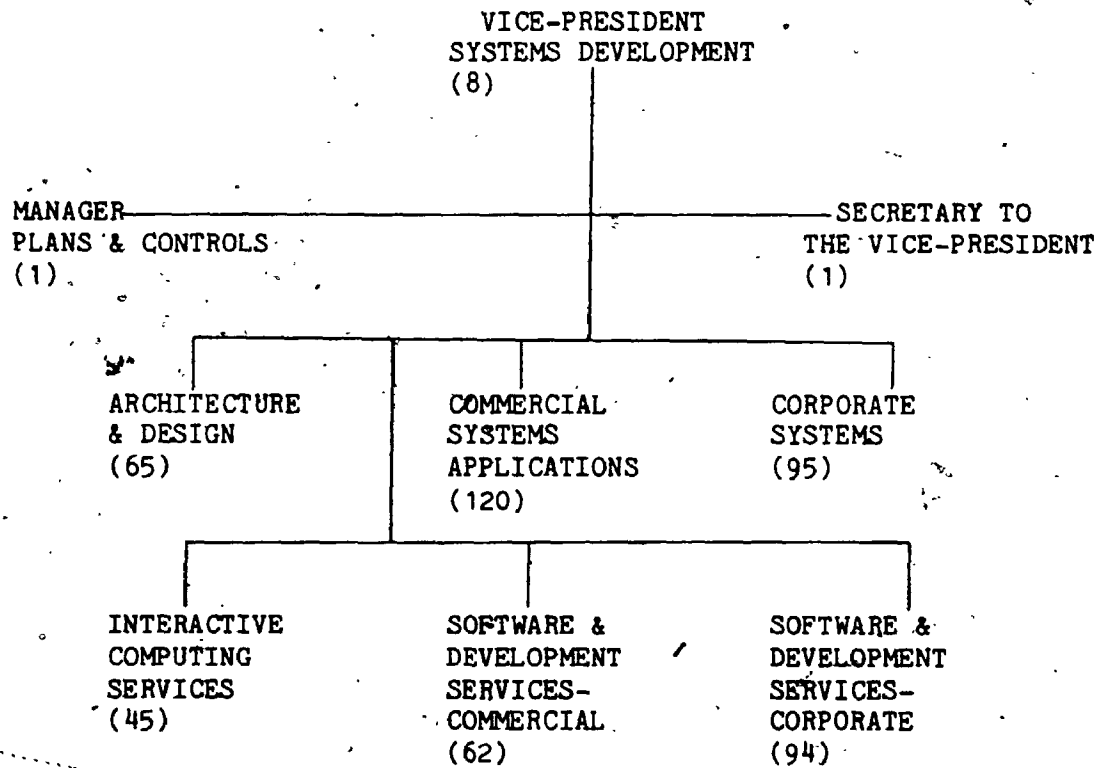
FIGURE G.1
DP ORGANIZATION AT G



specifications provided by the Architecture and Design group. Software and Development Services is also responsible for writing systems procedures which will be used by Operations.

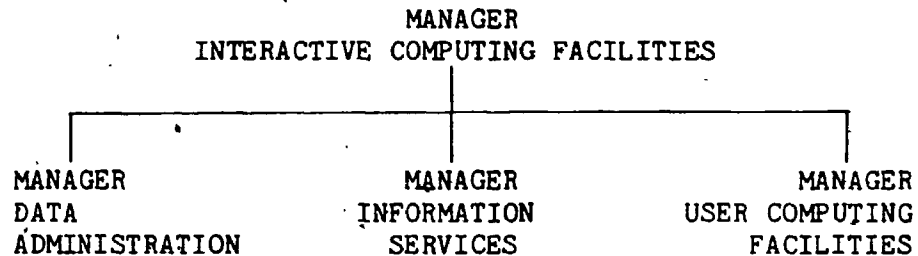
Interactive Computing Services is responsible for providing users with access to data for purposes of ad hoc reporting, and for supporting those users with the appropriate tools and techniques. Interactive Computing Services is composed of three sub-groups (Figure G.3): Data Administration, Information Services, and User Computing Facilities. Data Administration is mainly responsible for facilitating access to data by users while ensuring data security. Information Services is mainly responsible for developing small systems for users whereas User Computing Facilities provides tools and support to users who develop their own applications.

FIGURE G.2
SYSTEMS DEVELOPMENT ORGANIZATION AT G



Operations Group— Operations employs approximately 700 people, the majority of whom have duties directly related to the operation of the computers. In addition to the Computer Operations group, Operations is constituted by three sub-groups: Planning and Technical Services, Communications Facilities and Administration, and Financial Control. Operations processes more than three million transaction every day.

FIGURE G.3
INTERACTIVE COMPUTING FACILITIES AT G



2.3 DP from Inside

Approximately 94 percent of the DP budget is spent on development, operation and maintenance of operational control type of systems, whereas five percent is spent on management control systems, and one percent on planning systems. The five percent spent on management control systems represents the percentage of the budget spent on "pure" management control systems. That is, according to the Vice-president Systems Development, many of the operational control systems also have management control components. He suggests that the management control aspect of systems is more important than indicated by the percentages given above.

DP employs formal project management and project development methodologies, and standards exist for programming, systems operations, maintenance, and documentation.

If a group called Data Administration exists, its main task is oriented toward Data Base Administration. According to the

Vice-president Systems Development, there are ongoing discussions on the need to have a group responsible for "managing data as a corporate resource," but decisions on the location and the reporting relationships of such a group are yet to be made.

2.4 Relationships with the User Community

According to the Vice-president Systems Development, DP is highly visible at G. A Systems Steering Committee meets two hours per month, in order to review development plans, new development projects, etc. The Steering Committee is composed of the Deputy Chairman of the company, the President, the Executive Vice-presidents of all the company's North American divisions, the Senior Vice-president Corporate Planning, the Senior Vice-president Operations and Systems, and the Vice-president Systems. According to the latter, there is, from the most senior levels of user management, "a tremendous push toward automation" of the company's activities.

Most DP activities are centralized but ad hoc Information Systems groups have developed in some departments. DP explains the ad hoc groups as follows. Each division has a person (or a few people) who is the "contact" with the Architecture and Design group. This contact has the responsibility of the user's side of feasibility studies. Since the advent of UDA in the company, some of these contact groups have enlarged their responsibilities so as to include some application development.

Approximately 75 percent of the total DP budget is assessed to users. According to the Vice-president Systems Development, if it may happen that some users consider the DP assessments as "funny money," and do not pay as much attention as they should to these charges, the company's financial controllers "surely do not have the same attitude." If a department or a division DP expenditures are well above what was budgeted, the department or division will have to explain the discrepancy in the same way as if it was "real" money.

3. UDA at G

3.1 Background

At G, users who develop their own applications are supported by the User Computing Facilities group. User Computing Facilities is part of Interactive Computing Services, whose "raison d'etre" encompasses more than user support in the context of UDA. Interactive Computing Services has as its mission to "decentralize access of data for purposes of ad hoc reporting to end users by providing them with support tools and techniques" [1]. This mission is fulfilled by developing mini systems for users and by creating an environment where users can develop their own applications. A by-product of the activities of Interactive Computing Services is to relieve the main Systems Development group from some of its workload. One of the major reasons for creating Interactive Computing Services was the increasing demand for inquiry and analysis type of applications, combined with the inability of the main Systems Development group to satisfy the demand. The inability of Systems Development to satisfy the demand is explained

by the large backlog of production systems and by the fact that the applications referred to above do not have as strong business cases as production systems. Three groups participate to the fulfillment of Interactive Computing Services mission: (1) Data Administration, (2) Information Services, and (3) User Computing Facilities.

Data Administration. The Data Administration group is responsible for creating the environment for user access to data. The group is responsible for data base facilities evaluation, data base standards, logical data base design, and data management. Some of the key issues for this group are: "data security, data ownership, copy management, capacity planning, network architecture and planning, service contract (with Operations) for end-user programming" [2].

Information Services. The main responsibility of Information Services is to take some of the load off the main Systems Development group. Information Services actually develop applications for users. The criteria for an application to be developed by Information Services instead of being developed by the main Systems Development group are the following:

- one time request
- does not generate accounting data or transactions
- does not update production files
- is not shared by several departments
- is not critical to daily business
- does not impact production run schedules
- system taken over by user after development
- system maintenance and support is not time critical [3]

Information Services has been in place for approximately 5 years.

Examples of applications developed by this group are: marketing campaign monitoring, market research information, applications providing statistics for specified divisions, offices, etc. The main benefits due to Information Services activities are said to be the timeliness with which information is available to users, the low cost of providing the information, applications developed with minimal DP resources and increased levels of user confidence and interest in the services.

User Computing Facilities. User Computing Facilities is responsible for Management Support Services and Office Support Services. The Management Support Services group has the direct responsibility of supporting users in the context of UDA. The group has the responsibility of providing appropriate tools to users, training them, and providing assistance during the development of users' applications.

The group was created approximately two years ago. An important reason for making programming tools available to users was the increasing usage of outside timesharing services (approximately \$3 million in 1981). During the past year, the support group has grown from two people to six consultants and a manager.

3.2 UDA Now

As of December 1981, 35 users had received training with one or several of the programming tools made available through User Computing Facilities. Three months later, in March 1982, the number of trained users was 100. APL, ADRS-II, EASYTRIEVE, FOCUS, and IFPS are supported by the group. Both FOCUS and IFPS are new at G: IFPS became available

to users in December 1981, and FOCUS in January 1982. Classroom training is provided for ADRS-II and IFPS and will shortly be provided for FOCUS, while APL training is a self-taught computer-based course, and EASYTRIEVE is an audio-visual course followed by "hands-on" sessions.

The main formal restriction on the type of applications users can develop is that they cannot make changes to production files. Through copy management, users have access to copies of the portion of the production files for which they have authorization.

Three important objectives for the User Computing Facilities group, with regard to UDA, are: "encourage users do own coding, compete with external services bureaus, and assure that usage of service is cost effective."

(1) Encourage users do own coding. According to the Manager of User Computing Facilities, in the future users will be encouraged to develop their own ad hoc application rather than having them developed by Information Services, whenever it is suitable. An important reason for doing so is that Information Services has to face many requests for maintenance of the applications they develop for users. While, in theory, these applications should be one-shot type of applications and that maintenance should not be time critical, in practice, it is not always the case. Users re-use applications, sometimes with slight changes, which they ask Information Services to make. It is believed at G that users should be encouraged to develop themselves applications

for which appropriate tools are available. This would free Information Services to develop applications which require more DP expertise. In order to reach the objective of getting users do their own coding, it is seen as important to provide user friendly programming tools, "suitable to user objectives," provide appropriate training and consulting with the use of the tools, a "user friendly interface with the computer, and "an increased power of products."

(2) Compete with external service bureaus. Approximately one year ago a new policy has been put forth by the controller's department: This policy states that any new use of outside timesharing services should receive the approval of the Vice-president Systems Development. This approval will be granted if there is a business case to support the use of outside timesharing.

However, it is believed at G that for the policy to be effective in slowing the growth of outside timesharing usage, DP has to offer as good a service (if not better) as the service bureaus. The quality of service is most important in trying to convert outside timesharing users to in-house usage.

In order to compete with outside service bureaus, User Computing Facilities puts emphasis on: providing a "range of facilities suited to user, low terminal and communication cost, training resource, consulting resource, service availability and response, user friendly executive, willingness and ability to add new languages, packages, and use service bureaus terminals network access techniques" [4].

(3) Assure that the usage of the services is cost effective. Assurance that UDA is cost effective is ultimately the responsibility of users. However, according to the Vice-president Systems Development, when DP goes to the steering committee to obtain approval for increased capacity because UDA is widespread, DP will have the responsibility of demonstrating cost effectiveness. In order to assure cost effectiveness users pay direct costs where such costs apply (e.g. terminal rental), and receive an assesment of their use of DP resources. Users are also encouraged to provide written evidence of their successes in the context of UDA.

3.3 Success of UDA

According to the Vice-president Systems Development, evidence of the success of DP efforts to facilitate UDA is provided in the following ways. First, before creation of Interactive Computing Services, it often happened that senior management of the user community would ask DP to "lend" them a programmer to do some application development. Since users have been provided with programming tools, such requests no longer take place. Second, DP has been successful in repatriating a large proportion of outside timesharing usage. Third, the ability to assure the Steering Committee that UDA is cost effective is important in order to support DP requests for increased capacity due to usage made in the context of UDA. Every six months, the Vice-president Systems Development provides the steering committee with evidence of cost effectiveness. When Information Services develops mini systems they try to evaluate the savings due to the systems they develop. Users developing their own

applications are also encouraged to provide DP with evidence of their successes.

The decrease in the backlogs is not considered as being a critical measure of success for DP. First, the backlog is not actually measured. Second, as put by the Vice-president Systems Development, having users develop some of their own applications must have subtracted something from the backlog. "Yet, the backlog is infinite and even if you subtract something from infinity, you are still left with infinity." What has happened is that DP has been able to satisfy the segment of the user community who wanted to have access to data and could not have it through normal DP operations.

3.4 Concerns, Issues, and Problems

The main concern the Vice-president Systems Development has with regard to UDA pertains to the problems that may emerge when users develop applications which they would want to implement throughout the company, but which are not compatible with the systems architecture. If users wanted some of their applications to be implemented throughout the firm, these applications would have to be evaluated in the same way as any new project development, with regard to its fit with the systems architecture and prioritized on the basis of a business case. If, because they do not have the required characteristics, such applications did not receive DP approval for being part of the production systems, dissatisfaction might result from the part of users.

To the manager of User Computing Facilities, a key issue resides in the role of intermediary his group has to play between Operations and the user community. People in Operations are used to perform their scheduling on the basis of precise figures on capacity planning usually provided by Systems Development in the context of production systems: number of transactions per run, number of file openings, and so on. UDA does not easily lend itself to such planning, and the consequence is to disrupt the way Operations usually work. Moreover, production systems usually receive higher priorities when they are run. Because of that, it has happened that the response time was unsatisfactory to users. In order to resolve that issue, the manager of User Computing Facilities establishes a "service contract" with Operations. That is, they agreed on a response time which is both satisfactory from a user point of view and which does not create too much disruption for Operations.

According to the Vice-president Systems Development, "poor" capacity planning from the part of users has the potential to create problems for DP. Capacity planning for production systems is a task that DP performs satisfactorily; systems upgrades can be planned in advance and evidence can be provided for the need of such upgrades. Since capacity planning is not as easy in the case of UDA, it would be, as put by the Vice-president Systems Development, "difficult to go to the President and ask for \$6 million of increased capacity due to increase in application development by users." In order to justify, on an ongoing basis, the need for computer capacity, the Vice-president reports every six months to the steering committee some evidence of

cost justification of UDA. However, ultimately, the responsibility of cost justification belongs to the user community: if the Executive Vice-presidents, who are on the Steering Committee, approve computer usage in their own divisions, it is because they see cost justification.

3.5 The Future of UDA

The manager of User Computing Facilities has set three objectives for the group, namely, to growth in usage, regain the market for outside timesharing usage, and quality of service provided to the user community. The growth objective is to have 100 active users by April 1982 and 200 active users by the end of October 1982. Efforts will continue to be put on repatriating outside timesharing users, as well as on maintaining a high quality of service to the user community.

FOOTNOTES

[1] Excerpts from an internal document entitled: "An emerging DSS: A Practical Experience."

[2] Ibid.

[3] Ibid.

[4] Most users who are outside timesharing users use ASCII terminals, while the in-house computers use EBCDIC code.

COMPANY H

1. The Company

H is a firm manufacturing industrial products. The company's assets are of \$1.7 billion and its annual sales of \$1.4 billion. H employs 11,500 people, and is a wholly owned subsidiary of another company.

2. DP Profile2.1 Basic Facts

Computers have been used at H for more than 20 years, and a formal DP department was created in 1967, from the recommendation of a consultant's study. Computers are also used for process control; however, people working in that area, as well as the equipment used, are under the responsibility of engineering. DP hardware and communication monthly rental-equivalent is estimated to be \$250,000. Two computers are installed: an IBM 3031 which is used for running production systems and an IBM 4341 which is used by DP professionals for development and testing, and by users in the context of UDA.

2.2 DP Organization

The General Manager of DP at H reports to the Vice-president Administration who is also responsible for Employee Relations, Health Services, Plant Security, and Product Control. DP has 200 employees working in three main areas: Systems Services, Computer Services, and

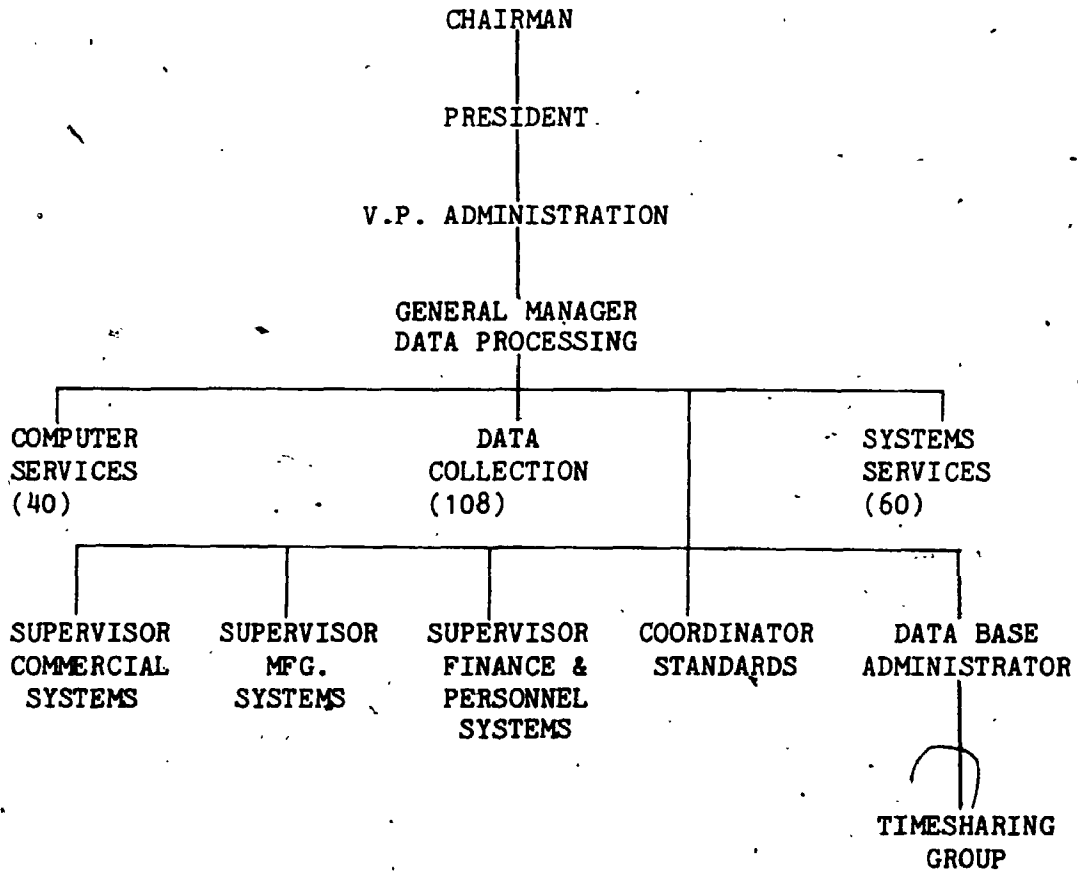
Data Collection (Figure H.1). Computer Services is responsible for systems programming, computer operations, hardware and software evaluation, and terminal network design. Systems Services has the responsibility for the development, implementation, and maintenance of systems, the determination areas where computer application is feasible, the assesment of computing requirements, data base administration, and timesharing services. Data Collection is responsible for the collection and editing of data, the dissemination of production and operating data, for taking and verifying physical inventories in the plants, and for the maintenance of inventory records. Data Colection is also responsible for Office Systems, which include printing, stationary, office furniture and equipment; and word processing.

2.3 DP from the Inside

The General Manager of DP estimates that approximately 90 percent of DP budget is spent on developing, operating and maintening operational control applications, while the remaining 10 percent of the budget is spent on management control applications. In the context of systems development, formal standards and methodologies have been established. There is no long range planning proceess in place for DP, but there ia a one year plan. A Data Base Administration group is in place, and has started some work on building a model of the organization data.

A major issue DP has had to face in the past years is the increasing dscrepancy between the demand and the supply for DP

FIGURE H.1
SYSTEMS DEVELOPMENT ORGANIZATION AT H

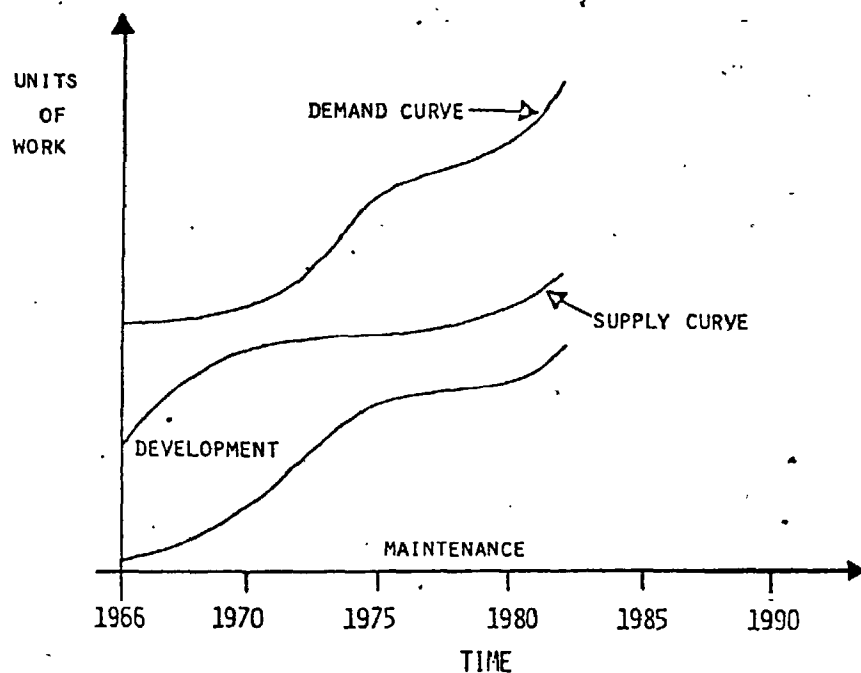


services. During the last year, twelve new people were hired within Systems Services but it is still insufficient to adequately respond to the demand. Figure H.2 outlines the relationship between demand and supply for DP services since 1966.

2.4 Relationships with the User Community

At H, a Steering Committee exists, which is constituted by representatives of the user community, at the level of Vice-presidents.

FIGURE H.2
EVOLUTION OF DEMAND AND SUPPLY OF DP SERVICES AT H



The chairman of the committee is the Vice-president Administration. The primary role of the committee is to review the priorities set by DP for large projects (such as timesharing for instance). Cost justification of projects is of major importance to the steering committee. Users are not charged for their use of DP resources: DP costs are treated as corporate overhead.

Some user departments (such as Accounting, Order Entry, Quality Control, and Product Control) have systems groups. The emergence of most of these groups is explained by the fact that there was a tendency to transfer an analyst who had been extensively involved in the development of a major system to the user department. Such a person is

knowledgeable of DP as well as of the user area; he could consequently play the role of intermediary between DP and the user department. Nowadays, the role of user systems groups is to design new systems down to "functional specifications," from which DP takes over. They also entirely develop some applications for users in their own department. People in these systems groups report to their own function and have no reporting relationship to DP. One exception is the case of a senior analyst in Marketing and Sales whose salary is paid by DP and whose main task is to identify requirements for application development in the area.

3. UDA at H

3.1 Background

In-house timesharing service was introduced in May of 1981. Yet, even before that date, UDA existed at H: several user departments were purchasing computer time from outside timesharing firms. Prior to 1977, outside timesharing usage was relatively low (less than \$10,000). However, from 1977 to late 1980, outside timesharing usage almost doubled each year. DP estimated that if the growth was going to continue at the same rate, outside timesharing usage would reach one million dollars by 1984.

Since DP knew that they did not have the resources required to do all application development for users, they considered the introduction of an in-house timesharing service. DP had previously recognized that many of the users' requests for applications did not fit "the way we

(DP) developed applications." That is, these requests were for data extraction, answering "what if" questions, and so on. One programmer was assigned to respond to these requests. In that context, two problems emerged quite rapidly. First, users would ask for several small changes to their applications, and second, the type of applications users requested became more sophisticated. This solution was then considered as not being the appropriate one.

In the late part of 1980, DP made a presentation to the steering committee to receive approval for the implementation of an inside timesharing service. During the presentation, emphasis was put on the dramatic growth of outside timesharing usage and on the discrepancy between the demand and supply of systems development resources. If the users of outside timesharing were to be repatriate, the cost for timesharing usage for the company as a whole would be substantially reduced. The project was approved on the basis of cost reduction.

3.2 UDA Now

The timesharing services group started its activities in May 1981 and is under the responsibility of the Data Base Administrator. The main reason for which timesharing is under the responsibility of this manager is that timesharing users will heavily interface with the database and DBA is the best source of knowledge about data location, codes and can ensure data security. Moreover, the present Data Base Administrator had shown keen interest in the concept of UDA. Most timesharing users are located in buildings that are remote from the location of the timesharing support group, and a large number have

terminals available at their own location. However, three terminals are available for users at the timesharing support group location.

The timesharing support group provides support for EASYTRIEVE, GRAPHICS, PLANCODE, and ADRS-II. One consultant is in charge of ADF, tool which is not actually used by users themselves. APL and FORTRAN are available to users but are not supported.

Timesharing concentrates its efforts on repatriating the outside timesharing users rather than on promoting its services to new users. However, as of March 1982, approximately two third of the in-house timesharing services were "new" users. At that date, a dozen of departments still used outside timesharing, and the cost for the month was \$50,000. Three departments accounted for almost 90 percent of the cost, one having spent almost \$25,000. The conversion of one of these users is planned for the fall of 1982.

According to the manager in charge of timesharing, conversion of the larger users of outside timesharing is likely to be slow. First, these users have access to large and specialized packages which are not available in-house and second, conversion of the programs and retraining of people would be necessary in some cases. However, it is felt by DP that given the current difficult economic situation, the high visibility of outside timesharing cost, and the support of top management for inside timesharing should have an important role to play in bringing in outside users.

With regard to new users, timesharing chose the avenue of "letting the word spread within the user population" rather than heavily promote the programming tools. The main reason for doing so is that DP did not want to see usage increasing so rapidly that users could not be adequately supported and that capacity problems would soon emerge. As of March 1982, the number of in-house timesharing users was approximately 150, number equal to what had been forecasted. Of these users, 40 to 50 actually develop applications, while the remainder mainly run applications developed by others.

There are no formal restrictions for timesharing users except that they have read-only access to production files. Additional disk space is made available to users when "they really need it."

Six consultants constitute the timesharing support group. Each one of the five consultants concentrates his/her activities on one programming tool and can provide back-up support for another tool. The sixth consultant is mainly responsible for administration, but he also supports the programming tools. Administrative tasks include control of users accounts and enforcement of guidelines for application development. To date, no formal guidelines exist; however, during the training period, users are made aware of certain practices which may create problems and are instructed about "dos and don'ts." Enforcing these guidelines means, for the Data Base Administrator, discussing with the user and trying to explain to him that the problems he faces are due to a misuse of the tools.

Support provided by the consultants includes training with the use of the tools (actual teaching for ADRS-II, PLANCODE and GRAPHICS; selftaught course for EASYTRIVE), providing advice on the choice of programming tools as well as debugging assistance. Consultants should not write code for users, but as put by one of them: "Sometimes, when you have tried everything else, as a last resort you will write a few lines of code."

Most user requests for assistance are made by phone calls, and according to consultants, immediate assistance can be provided right away for about 75 percent of these requests. Another 20 percent of the requests for assistance find a response within less than one hour. Timesharing consultants spend approximately 60 percent of their time in non-direct support activities, that is, developing courses, learning more about the programming tools, and so on.

3.3 Success of UDA

To top management, success of in-house timesharing will be measured by the reduction of outside timesharing usage, or in the short run, the reduction of its growth. It will be crucial to show this reduction when DP has to obtain approval for increased capacity for in-house timesharing usage. Consequently, as it was stated previously, DP concentrates its efforts on making this cost reduction become a reality.

The General Manager of DP identifies other benefits due to UDA. Before being in his present job he had moved from DP to a user

department. During the period where he was a user he realized that several activities were still performed manually and that it was effectively impossible to ask "what if" types of questions. The problems the department had to deal with were current and to have applications developed by DP would have meant years of waiting. At that time there was no in-house timesharing services; consequently, the department started dealing with a service bureau.

From this experience as a user, the DP manager identifies two main benefits of UDA. The first is the timeliness of application development. The second benefit resides in the fact that users no longer have to translate their needs to DP professionals; UDA dramatically reduces this "communication overhead."

DP encourages users to identify and report cost savings or other monetary benefits due to UDA. For instance, one department reported that for a given application: "The development costs were three man months and the benefits were 127 man hours per month." Excerpts of such "reports" are included in presentations made to the steering committee, so as to support the concept of UDA.

User satisfaction with the services provided by timesharing group is an important objective, both for the General Manager of DP and for the Data Base Administrator. In that case as well users are encouraged to report their experience with the assistance and support provided by the timesharing group.

According to the DP manager it is possible that requests for perfective maintenance will decrease. However, it is not considered as a critical measure of success of UDA and is not actually measured. It is not expected that the backlog will be reduced; it is currently very large, and there is a large latent demand. It is felt that applications users develop are either part of this latent demand, or are applications that were put on the lowest part of the DP priority list.

3.4 Concerns, Issues, and Problems

To the timesharing group a major concern is that users often try to develop applications that are "monsters." That is, users tend to develop applications which are too large in comparison with their knowledge of the tools, the experience they have with application development, the time they can devote to this activity, or their systems analysis skills.

The following scenario illustrates what often happens. In a department a supervisor becomes aware of the timesharing services. He decides to computerize applications which would save quite a lot of time to his people, for instance, a manual task that is performed by two or three people. The supervisor selects one individual to receive training with a tool. It may be a person who actually performs the job or somebody who is not very busy at that time. Generally, these individuals do not have much experience with programming and with systems analysis. Once the individual has been trained, he comes back to his department and has to computerize the application. If the

individual is also responsible for performing the manual job, he becomes overloaded and cannot concentrate on application development. If it is an individual who was not involved in the job previously, his lack of analysis skills and his lack of knowledge of the task have as a consequence that application development will be a task with which he cannot cope.

To solve this problem, the Data Base Administrator suggests that users start with small applications so that they learn a little at a time. The timesharing group also plans for providing users with an analysis course, since analysis is one of the weakest skills of most users. The situation described previously often happens at H. One department, when faced with the problem, decided to hire a programmer to do application development. It was felt, in that department, that the solution was satisfactory.

Accuracy of the information provided by the applications developed by users is another concern that the Data Base Administrator has, with regard to UDA. User applications are not audited and the support group is not, and cannot, be aware of all that users do. The Data Base Administrator does not see immediate solutions to this problem, that is, solutions which would not violate the philosophy of UDA. Packages with self-documentation and application development guidelines are considered as being elements of solution. However, as he and the DP manager mentioned, DP cannot enforce these guidelines. It is the responsibility of the management of each user area to put emphasis on control of applications their people develop.

According to the DP General Manager, UDA is seen as disruptive by people responsible for computer operations. They are used to, and have the responsibility for, scheduling jobs and keeping their environment as well controlled as possible. UDA asks for non-scheduled jobs, and because users are not programming experts their programs sometimes enter infinite loops. Moreover, users want fast response time. All of the above creates an environment that is quite different from the usual operations environment.

The DP manager also stated the issue of the computer being idle during the non-working hours. Users run their jobs when they are at the office, and at night the computer is not used. He hopes that when users are charged for their use of computing resources (as of 1983), that the lower cost of running jobs at night will incite users to have some of their programs run at night.

3.5 The Future of UDA

To date, usage that users make of the computer represents between 15 to 20 percent of the company's computing power. Hundred and fifty people use the timesharing services. It has been forecasted that by 1984 there will be approximately 600 timesharing users. It is also expected that by the same year, the timesharing requirements for computer power will be larger than the production systems requirements. At H, this increase of timesharing capacity requirements was explained by the following. First, UDA will grow because DP does not have the manpower for the development of all applications users want to be developed. Second, the tools provided to users are not as computer

efficient as the languages DP professionals use.

However, poor efficiency of tools is not a major concern, mainly because nowadays manpower costs are much higher in comparison with software costs.

COMPANY I

1. The Company

Company I operates in the communications industry. The firm employs 14,400 people, has assets of \$2.3 billion, and annual sales of \$900 million. I has its rates set by a regulatory agency, and is a wholly owned subsidiary of another company.

2. DP Profile

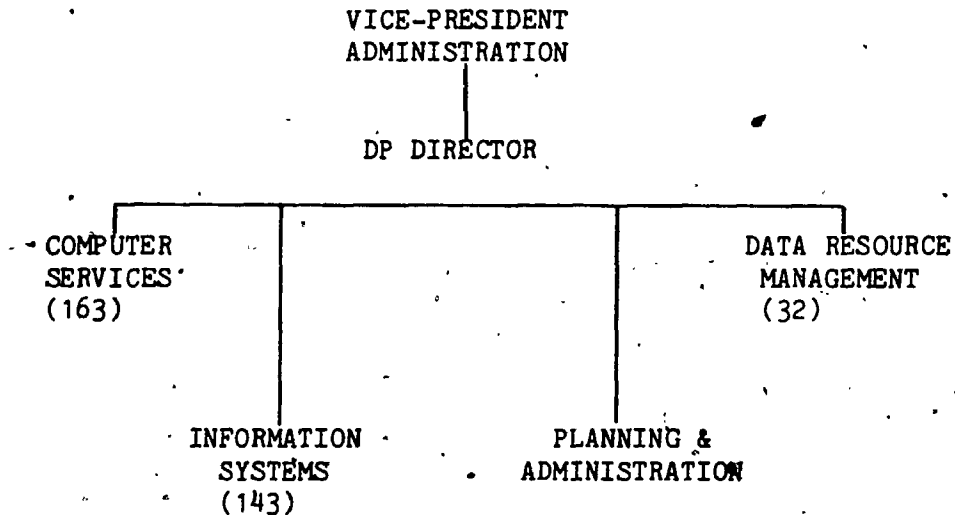
2.1 Basic Facts

Company I has been using computers for commercial DP for more than 20 years. DP expenditures for hardware and communications costs are nearly half a million dollars per month, on a rental equivalent. DP employs approximately 400 people. One IBM 3081 and one AMDAHL V8 are used mainly for running large systems such as Payroll, Customer Billing, Inventory, and so on. A DEC 10 is devoted to a special system, namely, the Service Order Update system, which was entirely developed by DP. Two DEC 10s and one IBM 4341 are devoted to UDA. Several mini computers are used in the company, and most of them were introduced by DP.

2.2 DP Organization

The Director of DP reports to the Vice-president, Administration. Four main services compose the DP department: Computer Service, Information Systems, Data Resource Management, and Planning and

FIGURE I.1
DP ORGANIZATION AT I



Administration (Figure I.1).

Computer Service is responsible for data entry, computer network, and technical services. Data Resource Management is responsible for data base administration, data administration and data communication. The group is in the early stages of data modeling. Planning and Administration is responsible for DP strategic planning, DP administration, DP performance measurement and control, and manpower planning and training.

Information Systems is composed of two specialized systems developments groups (the administrative group and the customer

information systems group) and of Systems Management and Support Service (Figure I.2). The latter is responsible for providing support services to the DP department and to the user community. Support to the DP department is provided through software engineering and systems management groups. The Information Centre provides support to users who develop applications. A Mini and Micro computers group provides assistance to both users and DP department people.

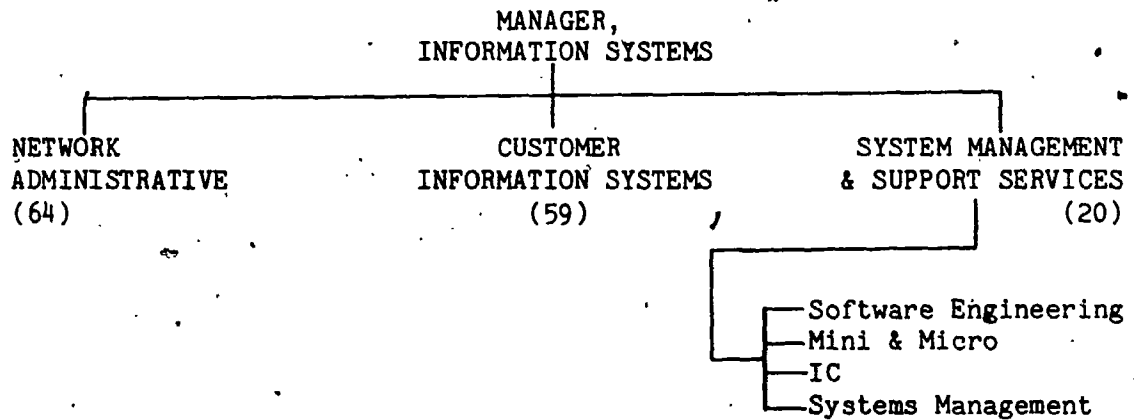
Several minicomputers are used in the company and most of them were brought in by the DP department. These minicomputers process applications which are "stand alone." The main reason for using minicomputers in such situations is that they are the least expensive solution for applications that do not require the power of the mainframe. The Mini and Micro group works in conjunction with systems development people to identify applications for which a minicomputer is the most appropriate solution.

It is also the responsibility of the Mini and Micro group to develop guidelines for the acquisition of mini and microcomputers. There exist user departments (6 which DP is aware of) that have acquired microcomputers without being advised to do so by DP. While the DP department is not against the acquisition of microcomputers per se, they want to develop guidelines for their acquisition.

2.3 DP from Inside

In the context of application development, DP uses formal standards for systems development and programming and is in the early

FIGURE I.2
INFORMATION SYSTEMS ORGANIZATION



stages of using a project management methodology. Approximately 69 percent of the DP budget is spent on operational control applications, 24 percent on management control applications, and 7 percent on planning systems. DP long range planning is performed by the Planning and Administration group.

2.4 Relationships with the User Community

At I, there exists a DP steering committee which is responsible for the approval of major DP projects. The committee is composed of the Chairman and CEO, the President and COO, and by the Vice-presidents. According to the DP Director, the most important criterion used by senior management in their evaluation of new development projects is the reduction of costs. Apart from the steering committee there exists a User Council, constituted by directors

of user departments. The User Council is a working group which has the responsibility of prioritizing new projects and making suggestions to the steering committee.

In general, users are heavily involved in new systems development activities, such as identifying new projects, determining the benefits of such projects, and designing the systems.

Except for CPU time when using in-house timesharing services, users are not charged for the use of DP resources. However, there is an ongoing project to examine chargeout schemes.

In some user departments there are groups of people who are heavily involved with application development. The degree of formality and the size of these groups vary according to the different departments. Cost, Prices and Economics, as well as Marketing have the most formal and the largest units of people devoted to application development. Other departments, such as Customer Services and Network Services have smaller units who are not formally recognized as systems groups. People who are members of these groups often are DP professionals who have been transferred from DP to user departments and who have used their expertise to build these groups. Some units were created by non-DP professionals who became interested in DP and who became more and more involved with application development. These groups have two main roles; first they provide specifications for applications that will be developed by the DP department and second they entirely develop some other applications. The latter are

generally kept within the department and run on the in-house timesharing or on outside timesharing. However, it happens that applications developed within user department are handed out to DP who runs them as production systems.

3. UDA at I

3.1 Background

Applications developed by users at I are run in one of the following three modes: they either use the in-house timesharing services, use outside timesharing services, or they use personal computers.

In-house timesharing falls under the responsibility of the Information Centre. Applications can be run on one of two DEC 10s, or on an IBM 4341 with CMS. FORTRAN, COBOL, and PASCAL are available on the DEC 10s while FORTRAN, COBOL, SAS, EASYTRIEVE, MARK IV, and SAS/GRAPH are available on the IBM 4341. Outside timesharing usage accounts for approximately one million dollars per year. There are six personal computers in users departments.

In-house timesharing has been available to users for approximately seven years. When it started an HP 2000 was used but it soon reached its capacity limits. It was then replaced by a DEC 10 which rapidly reached full capacity as well. A second DEC 10 was then acquired and during the same period DP started to charge timesharing users. Approximately two years ago the IBM 4341 with CMS was introduced.

In its early years, in-house timesharing was under the responsibility of technical services. In late 1977 an IC was created, with one supervisor and two staff people. Nowadays, the IC is staffed with 5 consultants and one supervisor. Two consultants are mainly responsible for IBM support, two are responsible for DEC support, and one consultant performs primarily administrative tasks.

The current DP manager explains the introduction of in-house timesharing in the following way. At I, the demand volume for application development has always "outstripped the DP resources available." Users became more sophisticated, both in their knowledge of computing and in their awareness of how the use of computers can support them in their jobs. Timesharing was available from outside firms and some user departments started to make use of it.

On the other hand, the individual who was president of the company at the time of the introduction of in-house timesharing believed that managers of individual departments were responsible for their own actions. According to the current DP manager, this philosophy was supportive of the idea that user departments could develop their own applications if they felt the need to do so and if they had the resources required.

In-house timesharing was introduced as a means of slowing the growth of outside timesharing rather than stopping it. To date, outside timesharing usage accounts for approximately one million dollars per year. It consists mainly in the use of very specialized

packages which fit specific problems and which would be much too costly to develop in-house. DP itself is a user of outside timesharing for a very specialized application.

3.2 UDA Now

Usage of in-house timesharing grew quite rapidly; in early 1979 there were approximately 500 registered accounts. In the spring of 1982 there were more than 1200 registered accounts. However, this does not mean that 1200 in-house timesharing users actually develop applications. First of all, one individual may have more than one account. According to the IC consultant in charge of administration, the 1200 accounts may represent 600 to 700 different individuals. Of these people, 450 are believed to be developing applications. The users run applications which were developed either by other DP or by other users. The consultant estimates that approximately 25 percent of the users who develop applications are sophisticated type of users (i.e. users who have a formal DP background or who have much experience with computers) while 75 percent are of the "DP amateur" type. Approximately 50 percent of all the IC users are in remote locations, that is, outside the head office. It is estimated that between 10 to 15 percent of the total CPU time is used by applications developed by users.

FORTTRAN and COBOL are the most heavily used programming tools. EASYTRIEVE was introduced in August 1981 and is not heavily used yet. SAS and SAS/GRAPH have been available for approximately two years and have not been extensively used. A tentative explanation provided by

the senior IC consultant is that until approximately eight months ago, there was no real training and support for these two tools. Eight months ago two consultants were hired, who knew SAS and who could support users.

The assistance provided by the IC includes training with the use of CMS, DEC, and SAS. Training with EASYTRIEVE consists in an audio-visual course. Users who want to learn FORTRAN, COBOL, or PASCAL are advised to take outside courses. The IC consultants advise users on the proper programming tools to use for a given application, and provide debugging assistance. It is a policy of the IC that consultants do not write code for users.

Most users have terminals available in their departments and there are two terminals available for users on the same floor as where the IC is located. In the near future there will be a drop-in centre with several terminals available. According to the IC consultants, users generally request and receive assistance over the telephone: approximately 80 percent of the requests for assistance are made by telephone.

End user meetings are held twice a year. During these meetings people from DP technical services make presentations about different hardware and software aspects, and some users present their experience with application development. In the future, the IC will try to have user departments nominate timesharing coordinators. User concerns, suggestions, grievances would be funnelled through these coordinators.

In the recent past, the IC has started going to user departments and making presentations on the services they offer.

Four of the five IC consultants have formal DP background and one come from the user community. While technical expertise is an important asset for IC consultants, much emphasis is put on their ability to relate well with people.

3.3 Success of UDA

According to the supervisor of the IC, up to now, the IC had to put most of its energy in keeping pace with the users, and it is felt "that this task has now been accomplished." For the future, the IC should concentrate on "anticipating the user needs, offering guidance for their future use of timesharing services, providing the necessary software products and accompanying support to meet their needs, and to sell the services of the IC."

Some of the more precise objectives for the coming year are "to make presentations to user groups, to develop and implement procedures to track measures of success, to publish an IC Standard Product Line, to provide a drop-in centre, to define the role and responsibilities of the timesharing coordinator and have individuals assigned representing all areas of the company."

To the IC supervisor, the "primary measure of success of the IC operations is the assurance that the end user community is reaping dollar benefits from their use of the timesharing facilities, and that

those benefits remain constant or increase relative to any expenditure." Some suggested measures of success are the cost benefit ratio and user satisfaction index (which would be obtained through a survey).

3.4 Issues, Concerns, and Problems

One of the DP manager's main concerns with respect to UDA is that there are applications that users develop which are critical to the company. In order to avoid such a thing, without imposing rigid controls which would be effectively impossible to enforce and much unwelcome from users, the DP manager feels that user managers should first be convinced of the risk of such a matter. Most other tentatives of control are likely to be ineffective. Closely related to the previous issue is the concern that it may happen that users misuse the computer. In that case as well, the DP manager argues that control is the responsibility of the user departments.

3.5 The Future of UDA

In the long term, the DP Manager sees the role of the DP department as being to develop the core systems for the company. Data will be available to users to develop their own applications.

COMPANY J

1. The Company

J is a company producing chemical and chemicals products. The company is divided in 10 "businesses" (e.g. chemicals, gas products), the Vice-presidents and General Managers of which report to Senior Vice-presidents of the company. Some activities of the company are strictly under the responsibility of each individual business, others (such as Computing) are under the corporate responsibility. J has a total of 5000 employees, assets of approximately \$900 million, and annual sales of \$800 million. J has a parent organization in the United States and has plants in all provinces of Canada.

2. DP Profile

2.1 Basic Facts

J has been using computers for more than 20 years. At J, the DP department is responsible for the development and operations of a commercial production systems of all the business, as well as it is responsible for telecommunications. Usage of computers for process control falls under the responsibility of the various plants and is not actually related to the DP department. Approximately 95 percent of the commercial DP systems are run on the central computer. RJE stations are located in the plants and some plants have on-line terminals as well. The remaining 5 percent of the systems are run using the parent company timesharing services.

DP monthly hardware and communications expenditures are approximately \$200,000 and DP employs 70 people. Computer power is provided by an IBM 4341 and an IBB370-148.

2.2 DP Organization

The DP Director reports to the Vice-president and Treasurer, who in turn reports to the Chairman and CEO (Figure J.1). The company's controller reports to the same Vice-president. DP is composed of three main groups: Operations, Systems Development, and Programming.

At J, DP activities are centralized; however, in two of the businesses, a DP professional is responsible for the liaison between the user community and DP. In addition to it, for each major system, there is a user who is an intermediary between DP and other users.

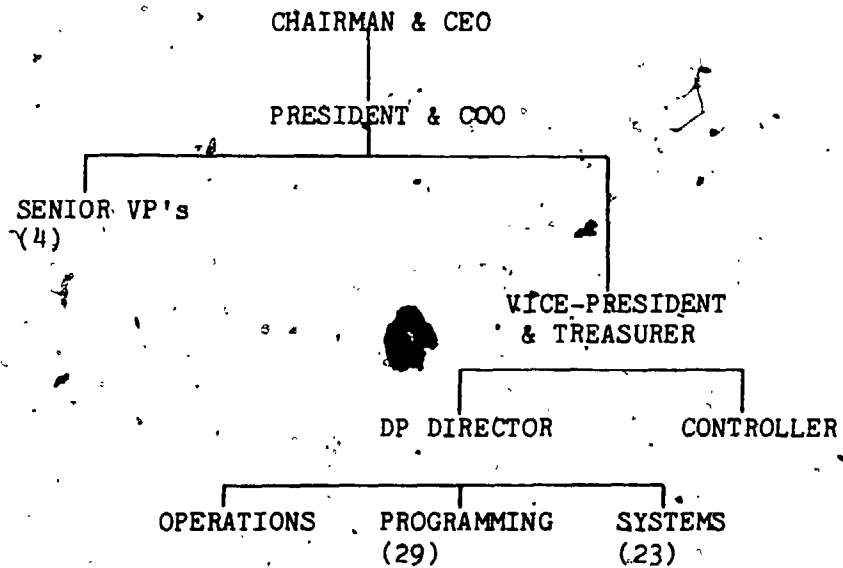
2.3 DP from Inside

Approximately 75 percent of the DP budget is spent on operational control systems, while management control systems account for about 20 percent and planning systems for the remaining 5 percent. Formal standards and methods are in use for systems development, and programming standards are under review. J has not established a Data Administration function but the data base technology has been introduced. The data base management system used is IDMS.

2.4 Relationships with the User Community

The DP department charges users for all their use of DP resources, and the charges are in real dollar terms. J does not have a DP

FIGURE J.1
DP. ORGANIZATION AT J



steering committee. A long range planning process is to be implemented, where the businesses will include the systems they want to be developed in the future. The plan will be approved by the Operating Committee, members of which are the Senior Vice-president and COO. Nowadays, DP is responsible for prioritizing requests for new systems; yet, requests for major systems are presented to top management on an ad hoc basis. The backlog of applications DP has to face is not large, and the DP Director explains it mainly by the current difficult economic environment. He argues that users have tightly control their budgets and they are charged for their use of DP resources. Expenditures on new DP projects are cut, as other expenditures are cut.

3. UDA at J

3.1 Background

Approximately 10 years ago, DP made some features of ASI-ST available to its users. The main reason for doing so was to cut small requests for report preparation. ASI-ST has file creation, file maintenance, data retrieval, data manipulation, and report generation features. However, users are not allowed to use the file creation and file update capabilities of ASI-ST.

For several years, a coordinator has been in charge of supporting ASI-ST users as well as training them. This task is a full time job, and the coordinator belongs to the Programming group of DP. ASI-ST users code their requests on special coding sheets provided by DP, and are responsible for having them key punched. The cards are read at the RJE stations for users in plants or sent to Operation for users at head office. During the last year, the demand for training with ASI-ST has increased more rapidly than in the previous years.

Some users use the parent company's timesharing services. The main languages used by J's users in that context are BASIC and IFPS. The latter was adopted approximately 18 months ago by J's Planning department. Six months ago, an analyst from the Systems group in DP was given the responsibility of supporting IFPS users. This new job came in conjunction with the decision to implement IFPS on the in-house computer, and to encourage IFPS users who are currently using services the parent company's timesharing to back in-house.

3.2 UDA Now

The ASI-ST coordinator estimates that, since the introduction of ASI-ST approximately 10 years ago, 200 users have been trained with it. However, a trained user is not necessarily an active user: some people have left the company, while others do not use the tool even if they have received training. According to this coordinator, the proportion of active users is low. He explains it by the fact that one has to practice quite a lot, after a two-day course, to become proficient with the tool. Since people have to go back to their duties after taking the course they often do not have time to practice and end up not using it.

While the ASI-ST coordinator is responsible for providing assistance on a full time basis, the IFPS coordinator provides support on a part time basis. The coordinator is also responsible for systems analysis duties. His main responsibility with regard to IFPS is to prepare the transfer in-house. There are approximately 30 IFPS users. Training with IFPS is not provided in-house, but either by Execucom, who markets IFPS or by DP consulting firms who provided such training. IFPS is mainly used for planning purposes. None of the tools are actively promoted.

IFPS users also receive support from the parent company which provides consulting services to its subsidiaries. Some IFPS users have been helped by these consultants, who assist in setting priorities with regard to which parts of the planning cycle should be "put on" IFPS

first, and who also provide assistance "if there is a problem."

3.3. Success of UDA

According to the DP Director, ASI-ST is successful. First, it appears that the number of small requests for reports has diminished since ASI-ST has been introduced. Second, and more importantly to this manager, from a cost/benefit point of view, ASI-ST provides high benefits to users, at low costs. The DP department does not track the benefits users get from UDA, since users have to justify the use they make of the computer to their own management.

3.4 Concerns, Issues, Problems

The Director of DP does not have major concerns with regard to UDA as it is now. In his opinion, UDA should be restricted to the area of personal computing; production types of applications should be developed in the controlled DP environment. According to this manager, the main difference between personal computing applications and production systems is a matter of size and of cost of development.

According to the ASI-ST coordinator, there are two main issues with end users using ASI-ST. The first is due to the fact that there are no editing procedures embedded within ASI-ST; hence, there may be errors in the information produced by ASI-ST applications. The second issue resides in the type of applications users try to develop. That is, some users develop applications which are much more complex than what ASI-ST is suited for, given that it is used by a non-DP person.

3.5 The Future of UDA

According to the Director of DP, in the years to come, the main role of the DP department will be to "provide the data bases," while users will extract and manipulate these data.

APPENDIX G
PROGRAMMING TOOLS

Below are short descriptions of the programming tools provided by DP in the firms studied. The main sources of information for those tool descriptions are vendors' publications. While BASIC, COBOL, FORTRAN, and PASCAL were made available to users in some firms, they are not described below.

APL: A Programming Language (APL) is a high-level interactive programming language which allows the user to perform simple to complex mathematical routines; create and update his own files, and produce reports. Because of the interactive nature and the power of the language, applications can be developed to produce meaningful results in less time than traditional programming languages. According to Crowder (1981), the characteristics which make a task a "natural" APL application are the following: interactive, quick and dirty one-shot problems where "the answer has tremendous value if computed 'right now', and no value if computed tomorrow," prototypes, and finally applications which are likely to be in constant evolution. APL being a partially interpretive language, one of its weaknesses is said to be run-time inefficiencies (Crowder, 1981).

APL/DI: VENDOR: IBM

APL/Data Interface (APL/DI) is a query language. Martin (1982) describes it as follows:

Its input can originate from IBM file and data-base systems. Information in multiple separate files can be linked together through common data.

End users are guided through the proper procedures by "prompt" messages at the terminal. Functions are provided for end users to select information and manipulate it in standard ways (i.e. summarize, cross-tabulate, count, sort,

accumulate statistics, etc.). Information can be selected using logical expressions (and, or, less than, equal to, etc.) or using computed values. (Martin, 1982, p.197)

ADRS-II: VENDOR: IBM

A Departmental Reporting System II (ADRS-II) is an APL-based product. It is an interactive tool, using an English like language. ADRS-II can generate reports either from production files or from data generated by the user. IBM claims that the tool was designed for business professionals and that it is easy enough to learn that the user can generate "customized reports after a few hours of training" (IBM, 1980). Some of ADRS-II main characteristics are its flexibility in designing report formats, as well as the availability of a large number of sort, selection, computation, and maintenance facilities. ADRS-II also has some graphic presentation capabilities, where user-selected data can be displayed in either point-graph or bar-chart formats. APL programs can be included within ADRS-II programs.

PLANCODE: VENDOR: IBM

There exist two versions of PLANCODE. The first one, Planning, Control, and Decision Evaluation System/Standard (PLANCODE/S) is the standard, batch oriented version. The second, PLANCODE/I is the interactive version. PLANCODE is a planning and budgeting tool which IBM describes as follows.

PLANCODE is an application program which allows executives, functional managers, and planners to use computers directly for:

- .Formulating and quantifying plans
- .Testing alternative courses of action
- .Monitoring actual activities and results versus planned results

PLANCODE/S is oriented towards the building of detailed plans and budgets for many departments and the consolidation of these different plans and budgets at divisional or corporate levels. In addition, PLANCODE/S enables the user to build a complete Budgetary Control application which supplies managers with the reports necessary for measuring actual results versus planned objectives....PLANCODE/I provides facilities for the rapid construction and implementation of planning models by means of an easy to-use plan simulation language. Facilities are included for consolidating plans and performing 'what if' type sensitivity analyses.

(IBM, 1979, p.3)

IFPS: VENDOR: Execucom

The Interactive Financial Planning System (IFPS) is an interactive modeling tool, which was designed for "non programming-oriented" users.

IFPS provides a natural language syntax for model (problem) formulation and a set of simple commands to solve the model and generate results. The problem statements are entered in any sequence desired. IFPS resolves the correct sequence of computation. The end result is, in the words of an IFPS user, that 'IFPS allows the user to focus on the problem, not on the solution vehicle.' Because the statements of the model are readable, they provide documentation for the problem solution. And because IFPS makes modification and/or interrogation of the model very easy, changes in assumptions or data take only a matter of minutes from conceptualization to implementation and report generation. (Execucom, 1979, p.1)

MODELER: MODELER is a model building tool which was developed in-house at C. It was developed primarily in response to the problems encountered in the development and use of planning models. These problems centered on the difficulty and cost involved in altering or expanding the tools employed in the planning effort. MODELER is

particularly suitable for planning and forecasting models. Equivalences may be defined and data transferred automatically between models with perfect freedom. MODELER also features automatic documentation, where all the variables involved in the model, the calculations used to derive them, as well as other information, are listed (Source: MODELER reference manual).

GIS: VENDOR: IBM

Generalized Information System (GIS) is a data base query language which is batch oriented. It allows users to select, sort, and list data from production data bases. In addition to those querying features GIS has facilities for logical and arithmetic operations, and its output can be edited and formatted as necessary. GIS can handle very large files. While several of the firms in the study make it available as a UDA tool, Martin claims that GIS needs "a specialist to use it, not a casual user without training" (Martin, 1982, p.130).

EASYTRIEVE: VENDOR: Pansophic

EASYTRIEVE is a batch oriented query language. In addition to selecting, sorting, and listing features, EASYTRIEVE provides for report editing and formatting capabilities. The tool also have features of high-level programming languages. Martin (1982) classifies it as being suitable for end users to use.

ASI-ST: VENDOR: Application Software Inc.

ASIS-ST is a batch oriented tool. It is a report generator as well as a file management system. It features a data dictionary, facilities

for selecting, sorting, and list data from files. It also performs arithmetic and logic operations. ASI-ST users at J fill in coding sheets, the content of which has to be punched on cards before being submitted to DP for execution. While ASI-ST has file updating capabilities, the feature to do so is not available to users at J.

FOCUS VENDOR: INFORMATION BUILDERS

FOCUS is a fourth generation language, which provides users with facilities such as data base querying, report preparation, graphics generation, high-level programming, statistical analysis capabilities, etc, through a single interface. FOCUS has its own DBMS but access to non-FOCUS data bases or files is also possible. FOCUS is claimed to be very easy to use.

SAS: VENDOR: SAS Institute

Statistical Analysis System (SAS) is primarily a statistical analysis package featuring frequencies, contingency tables, regression analysis, analysis of variance, and so on. SAS also provides for report writing, either simple or complex, graphing and bar charting, and file matching and merging.

SAS/GRAPH: VENDOR: SAS Institute

SAS/GRAPH is a graphics package fully compatible with SAS. SAS/GRAPH outputs may be directed to a wide array of display screens, printers, and plotters.

APPENDIX H

·REMINDER



The University of Western Ontario

DOCTORAL PROGRAM RESEARCH
School of Business Administration
London, Canada
N6A 3K7

REMINDER

RESEARCH PROJECT ON USER DEVELOPED COMPUTER-BASED APPLICATIONS

About two weeks ago you received a questionnaire concerning your experience with the development of computer-based applications. If you have sent in your completed questionnaire, please accept my sincere thanks. If not, I would appreciate your taking the time to do so.

The development of computer-based applications by users is relatively new. Meaningful research in this area is heavily dependent upon the information provided by users who have had experience with developing applications. Data gathered from the questionnaire will help better understand the context in which users develop applications. In addition, the questionnaire will serve to identify advantages associated with users developed applications as well as problems that may be encountered therein.

In the event that your questionnaire has been misplaced, please contact me and I will send you a replacement.

I would like to take this opportunity to thank you for taking part in this study and to emphasize that your cooperation is extremely important.

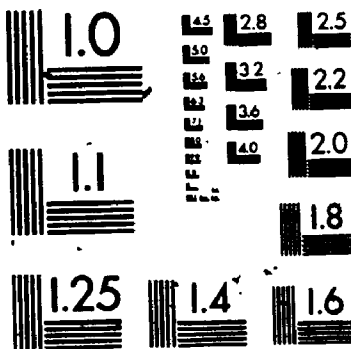
Sincerely,

Suzanne Rivard
Ph.D. candidate
(519) 679-6055

SR/rh

6 6

OF / DE



APPENDIX I
COMPOSITE SCALES

Composite scales were used to measure degree of Pull, degree of Push, user attitude toward UDA, user friendliness, satisfaction with support, satisfaction with independence, and satisfaction with environment setup. The items included in each scale are reproduced below.

PULL

I have often asked Data Processing to provide facilities (e.g programming tools, training, assistance) to users who want to develop applications (either for myself or for other users)

People in my department (other than myself) have often asked Data Processing to provide facilities to users who want to develop applications

PUSH

The Data Processing department is at odds with users developing their own applications

Data processing heavily promotes the idea of application development by users

The Data Processing department tries to facilitate access to the computer to users who want to develop applications

ATTITUDE TOWARD UDA

Colleagues are a better source of support than the Data Processing department people

I feel that I have the expertise required to develop the applications I do develop

Applications users develop themselves are often more useful than applications developed by Data Processing

For me, developing applications is a waste of time

I think it is not the job of users to develop applications

It is difficult to develop applications

USER FRIENDLINESS

For you, learning how to use this tool took	: A very short period of time	: ___ : ___ : ___ : ___ : ___	: A very long period of time
In order to use this tool	: Few keywords are required	: ___ : ___ : ___ : ___ : ___	: Many keywords are required
You have found these keywords	: Easy to remember	: ___ : ___ : ___ : ___ : ___	: Hard to remember
To you these keywords are	: Meaningful	: ___ : ___ : ___ : ___ : ___	: Meaningless
When you want to modify a program, the code is	: Easy to modify	: ___ : ___ : ___ : ___ : ___	: Hard to modify
This programming tool can be used for	: Many different types of tasks	: ___ : ___ : ___ : ___ : ___	: One type of task
The format in which you have to write a program is	: Flexible (keywords can be anywhere on a line)	: ___ : ___ : ___ : ___ : ___	: Rigid (e.g. specific columns for specific parts of a statement)
When there is an error in a program, the error messages are	: Easy to understand	: ___ : ___ : ___ : ___ : ___	: Hard to understand
You find the error messages	: Helpful in fixing errors	: ___ : ___ : ___ : ___ : ___	: Not helpful in fixing errors
Do you feel that	: The smallest error (e.g. a missing comma or an extra space) aborts a run	: ___ : ___ : ___ : ___ : ___	: Only severe errors will cause the program to stop running
The "help" commands on the system	: Are indeed helpful	: ___ : ___ : ___ : ___ : ___	: Are not helpful
The reference manual (or user manual) is	: Helpful	: ___ : ___ : ___ : ___ : ___	: Not helpful
Your background in computer programming was	: Sufficient to effectively use the tool	: ___ : ___ : ___ : ___ : ___	: Insufficient to effectively use the tool
To have the computer do a simple job	: A few instructions are sufficient	: ___ : ___ : ___ : ___ : ___	: Several instructions are required
In your knowledge, this tool could handle	: Large and complex types of jobs	: ___ : ___ : ___ : ___ : ___	: Small and simple types of jobs
For all the types of applications you develop, you use	: A single programming tool	: ___ : ___ : ___ : ___ : ___	: Many programming tools

SATISFACTION WITH SUPPORT

Quality of training (with the use of programming tools) you have received

Assistance provided by Data Processing in the choice of the proper programming tool for a given application

Assistance provided by Data Processing during the design of applications

Assistance provided by Data Processing in finding out and fixing errors in programs you write

Technical assistance provided by Data Processing

Knowledge of the programming tools by those who provide assistance to users

Understanding of the applications you develop by those who provide assistance to users

"People orientation" of those who provide assistance to users

Availability of those who provide assistance to users

SATISFACTION WITH ENVIRONMENT SET UP

Response time when you run an application

Availability of terminals

SATISFACTION WITH INDEPENDENCE

Not having to explain to a DP professional what an application should do

Reduction of development cost due to the fact that you develop the applications rather than asking Data Processing to develop them

The timeliness with which applications are developed (as compared to having them developed by Data Processing)

The control you have over the the applications you develop.

TABLE I.1
RELIABILITY COEFFICIENTS

<u>SCALE</u>	<u>NUMBER OF ITEMS</u>	<u>CRONBACH ALPHA</u>
IMPORTANCE OF SUPPORT	9	.747
IMPORTANCE OF ENVIRONMENT SET UP	2	.629
IMPORTANCE OF INDEPENDENCE	4	.535
SATISFACTION WITH SUPPORT	9	.873
SATISFACTION WITH ENVIRONMENT SET UP	2	.628
SATISFACTION WITH, INDEPENDENCE	4	.712
PULL	2	.620
PUSH	3	.728
USER ATTITUDE	6	.571
USER FRIENDLINESS	16	.757
WEIGHTED SATISFACTION WITH SUPPORT	9	.770
WEIGHTED SATISFACTION WITH ENVIRONMENT SET UP	2	.630
WEIGHTED SATISFACTION WITH INDEPENDENCE	4	.650

The scales were tested for reliability. Cronbach's Alpha was the reliability coefficient used. Table I.1 presents the results of the reliability tests. In this study, the reliability coefficients varied from .535 to .873. While reliability coefficients as high as .9 are said to be required in applied settings, reliability coefficients in the range of .5 to .6 are said to be sufficient in early stages of research (Nunnally, 1967, p.226; Peter, 1979, p.15). Consequently, the reliability coefficients obtained here were found to be satisfactory.

After the test for reliability, composite scores were computed for each scale. The method suggested by Hull and Nie (1979) to compute composite scores was adopted in this study. These authors suggest the following.

As a rule of thumb, it is often more useful to compute an average over the variables in the scale for each case rather than a simple sum. This has the advantage of allowing the interpretation of the scale scores in the appropriate metric of the items. For weighted composites this is not as appropriate, of course.

...to construct a composite score for a case with a missing value, the average value on the non-missing items is used as a scale score provided the case has missing data on fewer than 25% or so of the items.

(Hull and Nie, 1979, pp.84,85)

Two different types of composite scores were computed for satisfaction with support, satisfaction with environment setup, and satisfaction with independence. The first type is an unweighted score which calculated the average of the variables in the scale. The second type of score is a weighted score, where the importance score for a

given item was multiplied by the satisfaction score for the same item. The values obtained for all the items constituting a scale were then added. Below is an example for weighted satisfaction with environment setup.

	IMPORTANCE	SATISFACTION	WEIGHTED SCORE
Response time	4	3	12 (4x3)
Availability of terminals	1	5	5 (1x5)
			<hr/>
Weighted score for satisfaction with environment setup			17

BIBLIOGRAPHY

- Ackoff, Russell L., "Management Misinformation Systems", Management Science, Vol.14, No.4, December 1967; pp.147-156.
- Alloway, Robert M., "Temporary Management Systems: Application of a Contingency Theory to the Creation of Computer Based Information Systems," Unpublished Doctoral Dissertation, Harvard Business School, 1976.
- Alloway, Robert M., "Defining Success for Data Processing", Report CISR-52, Working Paper 1112-80, Sloan School of Management, M.I.T., March 1980a.
- Alloway, Robert M., "Users Managers' Systems Needs", Report CISR-56, Working Paper 1125-80, Sloan School of Management, M.I.T., May 1980b.
- Alloway, Robert M., Bullen, Christine V., Quillard, Judith A., "Users Needs Survey Research Project", Report CISR-73, Working Paper 1228-81, Sloan School of Management, M.I.T., June 1981.
- Alter, Steven L., "A Study of Computer Aided Decision Making in Organizations," Unpublished Ph.D. Dissertation, Sloan School of Management, M.I.T., 1975.
- Argyris, Chris, "Management Information Systems: The Challenge to Rationality and Emotionality," Management Science, Vol.17, No.6, February 1971, pp.275-292.
- Bariff, Martin, "Information Requirements Analysis: A Methodological Review," Working Paper 76-08-02, The Wharton School, University of Pennsylvania, January 1977, Revised December 1977.
- Bean, A.S., Neal, R.D., Radnor, M., Tansik, D.A., "Structural and Behavioral Correlates of Implementation in U.S. Business Organizations," in Schultz, R.I., Slevin, D.P., Implementating Operations Research/Management Science, American Elsevier, New York, 1975, pp.77-132.
- Beckhard, Richard, Harris, Reuben T., Organizational Transitions: Managing Complex Change, Addison-Wesley, Reading, Massachusetts, 1977.
- Benbasat, I., Dexter, A.S., Mantha, R.W., "Impact of Organizational Maturity on Information System Skill Needs," MIS Quarterly, Vol.4, No.1, March 1980, pp.21-35.

- Bennett, John L., "The User Interface Systems," Annual Review of Information Science and Technology, American Society for Information Science, Washington, D.C., pp.159-196.
- Boehm, Barry W., "Software and its Impact: A Quantitative Assessment," Datamation, Vol.19, No.5, May 1973, pp.48-59.
- Bradish, John R., "Administration of an Information Center User's Experience," Proceedings GUIDE 52, Atlanta, Georgia, 1981, pp.656-662.
- Buchanan, Jack R., Linowes, Richard G., "Understanding Distributed Data Processing," Harvard Business Review, Vol.58, No.4, July-August 1980a, pp.143-153.
- Buchanan, Jack R., Linowes, Richard G., "Making Distributed Data Processing Work," Harvard Business Review, Vol.58, No.5, September-October 1980b, pp.143-161.
- CODASYL, "A Status Report on the Activities of the CODASYL End User Facilities Committee," Information and Management, Vol. 2, No.4, 1979, pp.137-163.
- Codd, E.F., "Seven Steps to RENDEZVOUS with the Casual User," in Data Base Management, Klimbie, J.W., Kofferman, L., (eds.) North Holland Publishing Company, 1974, pp.179-200.
- Codd, E.F., "HOW ABOUT RECENTLY? (English Dialog with Relational Data Bases Using RENDEZVOUS Version 1)," in Schneiderman, Ben, (ed.), Databases: Improving Usability and Responsiveness, Academic Press, New York, New York, 1978, pp.3-28.
- Comper, F.A., "Experience with Information Centre," Proceedings GUIDE 51, Vol.2, Miami, Florida, November 1980, pp.1654-1665.
- Cooper, Randolph B., Swanson, E. Burton, "Management Information Requirements Assessment: the State of Art," Database, Fall 1979, pp.5-16.
- Chen, P., "The Entity -Relationship Model- Towards a Unified View of Data," ACM Transactions on Database Systems, Vol.1, No.1, 1976.
- Daly, Edmund B., "Organizing for Successful Software Development," Datamation, Vol.25, No.12, December 1979, pp.107-120.
- Data Processor, Vol.9, No.1, 1979.
- Data Processor, "The IBM Information Centre," Vol.11, No.3, 1981, pp.12-14.
- Date, C. J., An Introduction to Database Systems, Second Edition, Addison-Wesley, 1977.

- Davis, Gordon B., Management Information Systems: Conceptual Foundations, McGraw Hill, 1974.
- Davis, Gordon B., "Caution: User-Developed Decision Support Systems Can Be Dangerous to your Organization," Working Paper No. MISRC-WP-82-04, Management Information Systems Research Center, School of Management, University of Minnesota, 1981.
- Dearden, J., "MIS is a Mirage," Harvard Business Review, Vol.50, No.1, 1972, pp.90-99.
- Dertouzos, Michael L., Moses, Joel, (eds.), The Computer Age: A Twenty-Year View, The MIT Press, Cambridge, Massachusetts, 1979.
- Diebold, J., "Information Resource Management -The New Challenge," Infosystems, June 1979, pp.50-53.
- Dolotta, T. A., Bernstein, M.I., Dikson, R.S.Jr, France, N.A., Rosenblatt, B.A., Smith, D.M., Steel, T.B.Jr, Data Processing in 1980-1985, John Wiley and Sons, New York, New York, 1976.
- Doubler, J.L., Cooper, J.T., "What to Know About the Information Centre," Canadian Datasystems, April 1982, pp.76-80.
- Dubin, Robert, "Theory and Research," in O'Toole, Richard (ed.), The Organization, Management and Tactics of Social Research, Schenkman Publishing Company Inc., Cambridge, Massachusetts, 1971, pp.57-74.
- Eason, K. D., Damodoran, L., Stewart, T. F. M., "Interface Problems in Man-Computer Interaction," in Mumford E., Sackman (H.), (eds.), Human Choice and Computers, North-Holland Publishing Company, 1975, pp.91-105.
- Ein-Dor, Philip, and Segev, Eli, "Organizational Context and the Successes of Management Information Systems," Management Science, Vol.24, No.10, June 1978, pp.1064-1077.
- Execucom, IFPS Tutorial, Austin, Texas, June 1979.
- Ferguson, Cam, "Information Centre Users Experiences," Proceedings GUIDE 50, Houston, Texas, 1980, pp.1550-1554.
- Fried, L., "Centralization: 'To Be or Not to Be?'," Auerbach General Management Portfolio, No.1-01-01, Auerbach Publisher Inc., Pennsauken, New Jersey, 1976.
- Gibson, C.F., Nolan, R.L., "Managing the Four Stages of EDP Growth," Harvard Business Review, January-February 1974, pp.76-85.
- Ginzberg, Michael J., "A Detailed Look at Implementation Research," Report CISR-4, Working Paper 753-74, Sloan School of Management, M.I.T., 1974.

- Ginzberg; Michael J., "Implementation as a Process of Change: A Framework and Empirical Study," Report OISR-13, Working Paper 797-75, Sloan School of Management, M.I.T., 1975a.
- Ginzberg, Michael J., "A Process Approach to Management Science Implementation," Unpublished Ph.D. Dissertation, Sloan School of Management, M.I.T., 1975b.
- Glaser, Barney C., and Strauss, Anselm L., The Discovery of Grounded Theory, Aldine Publishing Company, Chicago, Illinois, 1970.
- Glaser, George, "The Centralization vs. Decentralization Issue: Arguments, Alternatives, and Guidelines," Data Base, Vol.2, No.3, 1970, pp.1-7.
- Goldstein, Robert C., McCririck, Ian B., "The Stage Hypothesis and Data Administration: Some Contradictory Evidence," Proceedings, Second International Conference on Information Systems, Cambridge, Massachusetts, 1981, pp.309-324.
- Greiner, Larry E., "Patterns of Organization Change," Harvard Business Review, Vol.45, No.3, pp.119-128, 1967.
- Grindlay, Andrew, "Information Centres," Business Quarterly, Vol. 45, No.4, Winter 1980, pp: 84-86.
- Grudnitsky, Gary, "A Methodology for Eliciting Information Relevant to Decision Makers," Proceedings of the Second International Conference on Information Systems, Cambridge, Massachusetts, December 1981, pp.105-120.
- Hoard, Bruce, "IBMer Calls Demand Processing Wave of Future," Computerworld, January 19, 1981, p.7.
- Holmes, Lauren L., "Information Centre -User's Signature," Proceedings GUIDE 51, Miami, Florida, November 1980, pp.1559-1582.
- Holton, John B., "Are the New Programming Techniques Being Used?," Datamation, Vol.23, No.7, July 1977, pp.97-103.
- Huff, Sid L., "Data Resource Management -Emerging Issues," Proceedings 1981 ASAC Conference, Information Systems Division, Halifax, Nova Scotia, May 1981.
- Hull, C.H., Nie, N.H., SPSS Update, McGraw-Hill, 1979.
- IBM, IBM System/370 Planning, Control, and Decision Evaluation System (PLANCODE) General Information Manual, Fifth Edition, GH19-1103-4, IBM Corporation, 1979.
- IBM, A Departmental Reporting System II, G309-0351, IBM Canada Ltd., December 1980.

- IBM, Information Centre Implementation Guide, Document No. GH09-0187, January 1981a.
- IBM, Business Systems Planning - Information Systems Planning Guide, Third Edition, IBM Corporation, 1981b.
- Iversen, Gudmund R., Norpoth, Helmut, Analysis of Variance, Sage University Paper series on Quantitative Applications in the Social Sciences, series no.07-007, Sage Publications, Inc., Beverly Hills, California, 1976.
- Jones, T.C., "The Limits of Programming Productivity," Proceedings Application Development Symposium, GUIDE-SHARE, Monterey, California, October 1979, pp.77-83.
- Kaufman, F., "Distributed Processing: A Discussion for Executives Traveling over Difficult Terrain," Data Base, Summer 1978, pp.9-13.
- Kenn, Peter G.W., "Computer-Based Decisions Aids: The Evaluation Problem," Sloan Management Review, Spring 1975, pp.17-29.
- Keen, Peter G.W., Scott Morton, Michael S., Decision Support Systems: An Organizational Perspective, Addison-Wesley, Reading, Massachusetts, 1978.
- Kennedy, M.H., Mahapatra, S., "Information Analysis for Effective Planning and Control," Sloan Management Review, Winter 1975, pp.71-83.
- King, William R., Cleland, David I., "The Design of Management Information Systems: An Information Analysis Approach," Management Science, Vol.22, No.3, November 1975, pp.286-297.
- Klein, Donald, "Some Notes on the Dynamics of Resistance to Change: The Defender Role," in Bennis, W.G. Benne, K.D., Corey, K.E., (eds.), The Planning of Change, Third Edition, Holt, Rinehart and Winston, 1976, pp.117-127.
- Kluckhohn, Clyde, "Common Humanity and Diverse Cultures," in Lerner, Daniel (ed.), The Human Meaning of Social Sciences, The World Publishing Company, Cleveland, Ohio, 1962, pp.245-284.
- Kolb, David A., Frohman, Alan L., "An Organization Development Approach to Consulting," Sloan Management Review, Vol.12, No.1, Fall 1970, pp.51-65.
- Kraft, P., Programmers and Managers, Springer-Verlag, New York, New York, 1977.
- Lehman, J.H., "How Software Projects are Really Managed," Datamation, Vol.25, No.1, January 1979, pp.119-129.

- Lientz, B.P., Swanson, E.B., Tompkins, G.E.; "Characteristics of Application Software Maintenance," Communications of the ACM, Vol.21, No.6, June 1978, pp.466-471.
- Lientz, B.P., Swanson, E.B., Software Maintenance Management, Addison-Wesley Publishing Company, Reading, Massachusetts, 1980.
- Lorsch, Jay, Kotter, John, Managing Change, Part I: The Problem of Resistance, Harvard Business School Paper 9-476-102, 1976.
- Lucas, Henry C.Jr., Toward Creative System Design, Columbia University Press, New York, New York, 1974.
- Lucas, Henry C.Jr., The Implementation of Computer-Based Models, National Association of Accountants, New York, New York, 1976.
- Lucas, Henry C. Jr., "Empirical Evidence for a Descriptive Model of Implementation," MIS Quarterly, June 1978, pp.27-41.
- Magee, John, "Consolidated Edison Information Center Timesharing Support," Proceedings GUIDE 51, Miami, Florida, November 1980, pp.1348-1360.
- March, James G., "Footnotes to Organizational Change," Administrative Science Quarterly, December 1981, pp.563-577.
- Martin, James, Design of Man-Computer Dialogues, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1973.
- Martin, James, Computer Data-Base Organization, Second Edition, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1977.
- Martin, James, Application Development without Programmers, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1982.
- Martin, T.H., Carlisle, J.H., Tren, S., "The User Interface for Interactive Bibliographic Searching: An Analysis of the Attitudes of Nineteen Information Scientists," Journal of the American Society for Information Science, Vol. 24, No.2, 1973, pp.142-147.
- Martin, T., "Information Retrieval," in Green, T.R., Smith, H.T., (eds.), Human Interaction with Computers, Academic Press, London, 1980, pp.161-175.
- McCracken, Daniel D., "Software in the 80s: Perils and Promises," Computerworld, Vol.14, No.38, September 17 1980, pp.5,9,10.
- McCricrick, Ian B., Goldstein, Robert C., "What do Data Administrators Really Do?," Datamation, August 1980; pp.131-134.

- McCririck, Ian B., "A Survey of the Data Administration Function in Large Canadian Organizations," Unpublished Masters Thesis, Faculty of Commerce and Business Administration, The University of British Columbia, June 1979.
- McKeen, James D., "Activity Analysis: An Approach to Understanding the System Development Process," Proceedings of 1982 ASAC Conference, Information Systems Division, Ottawa, Canada, May 1982, pp.41-50.
- McLean, Ephraim R., "End Users as Application Developers," MIS Quarterly, Vol.3, No.3, December 1979, pp.37-46.
- McNurlin, B.C., "Supporting End User Programming," EDP Analyzer, Vol.19, No.6, June 1981.
- Merton, Robert K., Social Theory and Social Structure, The Free Press, Glencoe, Illinois, 1958.
- Mintzberg, Henry, "The Manager's Job: Folklore and Fact," Harvard Business Review, Vol.53, No.4, July-August 1975, pp.49-61.
- Mintzberg, Henry, "An Emerging Strategy of 'Direct' Research," Administrative Science Quarterly, Vol.24, No.4, December 1979, pp.582-589.
- "Missing Computer Software," Business Week, September 1, 1981, pp.46-51.
- Munro, Malcolm C., and Davis, Gordon B., "Determining Management Information Needs: a Comparison of Methods," MIS Quarterly, July 1977, pp.55-67.
- Mumford, Enid, "Participative System Design: Structure and Method," Systems, Objectives, Solutions, Vol.1, No.1, January 1981, pp.5-19.
- Narasimhan, Ram, Schroeder, Roger G., "An Empirical Investigation of Implementation as a Change Process," TIMS Studies in the Management Sciences, Vol.13, North-Holland Publishing Company, pp.63-83.
- Nie, N. H., Hull, C. H., Jenkins, J. G., Steinbrenner, K., Bent, D. H., SPPSS: Statistical Package for the Social Sciences, Second edition, McGraw Hill Inc., 1975.
- Nolan, Richard L., "Computer Data Bases: the Future is Now," Harvard Business Review, September-October 1973, pp.98-114.
- Nolan, Richard L., "Managing the Computer Resource: A Stage Hypothesis," in Nolan, Richard L. (ed.), Managing the Data Resource Function, West Publishing Company, 1974a, pp.47-61.

Nolan, Richard L., and Seward, Henry H., "Measuring User Satisfaction to Evaluate Information Systems," in Nolan, Richard L. (ed.), Managing the Data Resource Function, West Publishing Company, 1974b, pp.253-275.

Nolan, Richard L., "Restructuring the Data Processing Organization for Data Resource Management," Proceedings of the IFIP International Conference, 1977, pp.261-265.

Nolan, Richard L., "Managing the Crisis in Data Processing," Harvard Business Review, Vol.57, No.2, March-April 1979, pp.115-126.

Norton, David P., "Information Systems Centralization: The Issues," Harvard Business School Paper, 9-172-286, 1972.

Norton, David P., "Organizing for the Computer: To Centralize or not to Centralize," in Nolan, Richard L., (ed.), Managing the Data Resource Function, 1974, pp.187-205.

Nunnally, J., Psychometric Methods, McGraw-Hill Book Co., New York, New York, 1967.

Oppenheim, A. N., Questionnaire Design and Attitude Measurement, Basic Books Inc., New York, New York, 1966.

Peter, Paul J., "Reliability: A Review of Psychometric Basics and Recent Marketing Practices," Journal of Marketing Research, Vol.16, February 1979, pp.6-17.

Petersson, L., "The Information Centre -User Experience," Proceedings GUIDE 53, Dallas, Texas, November 1981, pp.738-756.

Radnor, M.A., Rubenstein, A.H., Tansik, P.A., "Implementation of Operations Research and R & D in Government and Business Organization," Operations Research, Vol.18, No.6, February 1970, pp.967-999.

Read, Nigel S., Harmon, Douglas L., "Assuring MIS Success," Datamation, February 1981, pp.109-120.

Rivard, S., Grindlay, A., Huff, S., Suttie, P., "Decision Support Systems in Canada," Working Paper No.253, School of Business Administration, The University of Western Ontario, October 1980.

Rockart, John F., "Centralization vs. Decentralization of Information Systems a Preliminary Model for Decision Making," GUIDE, Washington, D.C., 1976.

Rockart, John F., Flannery, Lauren S., "The Management of End User Computing," Proceedings of the Second International Conference on Information Systems, Cambridge, Massachusetts, December 1981, pp.351-363.

- Rosenberger, Robert B., "The Information Center," Proceedings GUIDE 51, Miami, Florida, 1980, pp.1521-1558.
- Rosenberger, Robert B., "The Information Center," Proceedings GUIDE 52, Atlanta, Georgia, 1981a, pp.646-655.
- Rosenberger, Robert B., "The Productivity Impact of an Information Center on Application Development," Proceedings GUIDE 53, Dallas, Texas, November 1981b, pp.918-932.
- Runyan, Linda, "Software Still a Sore Spot," Datamation, Vol.27, No.3, March 1981, pp.165-167.
- Rudawitz, L.M., Freeman, P., "Client-Centered Design: Concepts and Experience," Systems, Objectives, Solutions, Vol.1, No.1, January 1981, pp.21-32.
- Sackman, H., Erickson, W., Grant, E., "Exploratory Experimentation Studies Comparing Online and Offline Program Performance," Communications of the ACM, Vol.11, No.1, January 1968, pp.3-11.
- Sammet, Jean E., Programming Languages: History and Fundamentals, Prentice Hall, Englewood Cliffs, New Jersey, 1969.
- Schein, Edgar H., "The Mechanisms of Change," in Bennis, W.G., Benne, K.D., (eds.), The Planning of Change, Holt Rinehart Winston Inc., 1969, pp.98-107.
- Schneiderman, Ben, Software Psychology, Winthrop Publishers Inc., Cambridge, Massachusetts, 1980.
- Schussel, George, "Data Bases," in McFarlan, F.W., Nolan, R.L., (eds.), The Information Systems Handbook, Dow Jones-Irwin Inc., Homewood, Illinois, 1975, pp.659-685.
- Seaman, J., "Microcomputers Invade the Executive Suite," Computer Decisions, Vol.13, No.2, February 1981, p.68.
- Secret, Richard D., "The Data Administrator," in McFarlan, F.W., Nolan, R.G., (eds), The Information Systems Handbook, Dow Jones-Irwin Inc., Homewood, Illinois, 1975, pp.647-658.
- SHARE, "Data Administration Project: Position Paper," (August 1980)
- Sheperd, A.J., "A British Example of Distributed Computing," Datamation, March 1978, pp.87-91.
- Siegel, Sidney, Nonparametric Statistics, McGraw-Hill, New York, New York, 1956.
- Simon, Julian L., Basic Research Methods in Social Sciences, Random House, New York, New York, 1978.

- Sircar, Sumit, "Directions in Distributed Data Processing," The Business Quarterly, Winter 1979, pp.79-85.
- Smith, H.T. "Human-Computer Communication," in Green, T.R.G., Smith, H.T., (eds.), Human Interaction with Computers, Academic Press, London, 1980, pp.5-38.
- Sorensen, Richard E., Zand, Dale E., "Improving the Implementation of OR/MS Models by Applying the Lewin-Schein Theory of Change," in Schultz, R.L., Slevin, D.P., (eds.), Implementing Operations Research/Management Science, American Elsevier Publishing Co., New York, New York, 1975, pp.217-237.
- Stork, Lauren L., and Bland, Charles R., "Control of Enduser Programming in Large Corporations: A Case Study," Unpublished Masters Thesis, Sloan School of Management, M.I.T., 1980.
- Swanson, E.B., "Management Information Systems: Appreciation and Involvement," Management Science, Vol.21, No.2, October 1974, pp.178-188.
- Swanson, E.B., "Measuring User Attitudes in MIS Research: a Review," OMEGA, Vol.10, No.2, 1982, pp.157-165.
- Taggart, W.M. Jr, Tharp, M.O., "A Survey of Information Requirements Analysis Techniques," Computing Surveys, Vol.9, No.4, December 1977, pp.274-290.
- Talbot, Jean, "Conceptual Data Modeling - A Tool for the Data Administrator," Proceedings of 1982 ASAC Conference, Information Systems Division, Ottawa, Canada, May 1982, pp.10-20.
- Thirkell, Peter, "Consumer Expectations Disconfirmation and Satisfaction," Unpublished Ph.D. Dissertation, School of Business Administration, The University of Western Ontario, 1980.
- Van Rensselaer, C., "Centralize? Decentralize? Distribute?," Datamation, April 1979, pp.89-97.
- Walton, Richard E., "Advantages and Attributes of the Case Study," The Journal of Applied Behavioral Science, Vol.8, No.1, 1972, pp.73-78.
- Waltz, David L., "An English Language Question Answering System for a Large Relational Database," Communications of the ACM, Vol.21, No.7, July 1978, pp.526-539.
- Weizenbaum, Joseph, Computer Power and Human Reason, W.H. Freeman and Company, San Francisco, California, 1976.
- Withington, Frederic G., "Coping with Computer Proliferation," Harvard Business Review, Vol.58, No.3, May-June 1980, pp.152-164.

Zaltman, Gerard, Duncan, Robert, Holbek, Jonny, Innovations and Organizations, John Wiley & Sons, New York, New York, 1973.

Zimbardo, P.G., Ebbesen, E.B., Maslach, C., Influencing Attitudes and Changing Behavior, Second Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1977.

Zloof, Moshe M., "Design Aspects of the Query-By-example Data Base Management Language," in Schneiderman, B., (ed.), Databases: Improving Usability and Responsiveness, Academic Press, New York, New York, 1978, pp.3-28.

END

1 | 6 | 0 | 9 | 8 | 3

FIN