Advances in COMPUTERS

EDITED BY

MARSHALL C. YOVITS

VOLUME 21

Advances in COMPUTERS

EDITED BY

MARSHALL C. YOVITS

Purdue School of Science
Indiana University–Purdue University at Indianapolis
Indianapolis, Indiana

VOLUME 21

1982

ACADEMIC PRESS

A Subsidiary of Harcourt Brace Jovanovich, Publishers

New York ■ London

Paris ■ San Diego ■ San Francisco ■ São Paulo ■ Sydney ■ Tokyo ■ Toronto

ACADEMIC PRESS, INC.
111 Fifth Avenue, New York, New York 10003

United Kingdom Edition published by
ACADEMIC PRESS, INC. (LONDON) LTD.
24/28 Oval Road, London NW1 7DX

LIBRARY OF CONGRESS CATALOG CARD NUMBER: 59-15761

ISBN 0-12-012121-2

PRINTED IN THE UNITED STATES OF AMERICA

82 83 84 85 9 8 7 6 5 4 3 2 1

Contributors to Volume 21

Numbers in parentheses indicate the pages on which the authors' contributions begin.

SUBRATA DASGUPTA, *Department of Computing Science, University of Alberta, Edmonton, Alberta T6G 2H1, Canada (91)*

ROBERT C. GAMMILL, *Mathematical Sciences Department, North Dakota State University, Fargo, North Dakota 58105 (155)*

ROB KLING, *Department of Information and Computer Science and Public Policy Research Organization, University of California at Irvine, Irvine, California 92717 (1)*

PETER LYKOS, *Department of Chemistry, Illinois Institute of Technology, Chicago, Illinois 60616 (275)*

JAMES E. RUSH, *James E. Rush Associates Incorporated, Powell, Ohio 43065 (333)*

GIOVANNI MARIA SACCO, *Database Systems Research Center, College of Business and Management, University of Maryland, College Park, Maryland 20742 (225)*

WALT SCACCHI, *Department of Computer Science, School of Engineering, University of Southern California, Los Angeles, California 90007 (1)*

S. BING YAO, *Database Systems Research Center, College of Business and Management, University of Maryland, College Park, Maryland 20742 (225)*

Preface

Volume 21 of *Advances in Computers* continues the publication of in-depth articles of substantial current interest to computer and information science. Subjects have been included not only because of their timeliness but largely because they are expected to be of interest over a considerable period of time. The publication provides an opportunity for highly regarded experts in their fields to write review or tutorial articles covering an important aspect of their field. By its very nature *Advances in Computers* permits the publication of survey articles of some length and with a perspective obtained by relatively leisurely preparation. Volume 21 thus continues a long and unbroken series that began in 1960. Many of the articles which have been included in the previous volumes have had a significant impact on the directions in which computer and information science and technology have developed.

Included in Volume 21 are contributions on applications, on some basic conceptual concerns, and on social and organizational impact on computing use.

In the first article, Rob Kling and Walt Scacchi point out that most scholarly and professional examinations of the social and economic repercussions of new computing developments have so far been based on a highly simplified conception of computing in social life. In these analyses the social context in which the technology is developed and used as well as the history of participating organizations are generally ignored for the sake of simplicity. It is also assumed that computing developers and users are rational decision-makers and are generally concerned with issues that are neither very short term nor based on narrow interests. These assumptions, they point out, are greatly oversimplified. Kling and Scacchi have proposed a "web model of computing" which emphasizes the way in which some focal computing resource is produced by a network of producers and consumers. According to web models, the macrostructures of broader social relations and the local infrastructure shape the kind of computer-based service made available. These are dependent upon a set of historical commitments. Computing developments are in fact complex social objects constrained by their context, infrastructure, and history.

Subrata Dasgupta indicates that few areas of thought have been so dependent upon language or notation as computer science. Further, this dependency ever increases as the field evolves and grows. It has become evident that the real issue in programming language design is not so much one of communicating with machines as it is of specifying computations in the most exact and appropriate manner possible. In programming, Dasgupta claims the power of languages transcends any ability to express ideas. Languages have helped to shape our thoughts on computation, to direct them along one path rather than another. He discusses the nature of languages that may be used for the design and description of computer hardware systems, with special focus on architecture descriptions as well as on the description of microprograms. Although, as he points out, there exist a large number of language proposals for hardware specification at various levels of abstraction, they have not generally been widely adopted. He believes that further applicability of these languages will depend to a large extent on the development of appropriate language-based design methods and automated systems. He states that the next decade will see a shift in emphasis from language design issues to the problem of methodologies, systems, and environments for the systematic, rigorous, and verifiable design and implementation of computer architectures.

Robert Gammill points out that microcomputers are rapidly becoming a pervasive element of our lives. By the end of the 1980s there will be few machines that will not contain a microprocessor, including many that we do not presently think of as machines, such as our homes. In his article, Gammill treats microcomputer applications, not only describing the present and extrapolating the future but also evaluating these applications under various criteria of suitability and acceptability. It is obviously impossible to discuss all areas in which microcomputers play a major role, since they will shortly be involved in virtually all aspects of human activity. But Gammill does treat a very wide spectrum of applications including hobby computing, business, industry, science, the home, agriculture and rural life, and communications. He concludes that the era of public use of microcomputers is just beginning and the promise is great. There are problems ahead and all citizens will need to educate themselves about computers. He also concludes that the future of microcomputers and digital communications is interconnected, and it is only through the joint success of each that the true promise of this new era will be realized.

In the fourth article data management is discussed by Giovanni Maria Sacco and S. Bing Yao. They remind us that data management has always been a major computer activity. Earlier data management functions were customized for particular applications, often being no more than elementary file management systems. This approach was expensive and inefficient, particularly as the amount of data grew rapidly. They state that data base systems were introduced to solve many of these problems. These systems offer to the user high-level primitives for data management and automatically perform many basic functions. Distributed data bases are considerably more complex than centralized ones and the field is still at its beginnings. Their article focuses on the problem of query optimization in distributed data base systems, and basic concepts are presented. Current methods for centralized systems are briefly reviewed. Distributed query processing is characterized and several major query evaluation strategies are discussed in detail followed by a presentation of a general framework for query optimization. They conclude in their article that problems remain to be solved such as dependence on network topology, initial materializations, various cost functions, the system load, and various objective functions.

Chemistry is an experimental science which uses the language of mathematics augmented by graphics. It involves making measurements, data reduction, transformation and representation, mathematical modeling and prediction, and information storage and retrieval. The chemist, Peter Lykos points out, is adaptable. He is accustomed to adapting new tools offered by mathematicians, physicists, and engineers and bringing them into his laboratory. Computers are seen as opening up completely new possibilities for doing chemistry. The chemist in adopting computers and their methodology thus becomes more potent and discovers new channels of communication with people working in other disciplines. Lykos considers several important applications of computers used in chemistry. These are education, laboratory automation, modeling, data bases, and the current chemistry establishment. Lykos concludes that the application of computers by chemists for chemistry is entering a period of very rapid growth. In addition, chemists are becoming increasingly important to computer designers and vendors as a source of new design ideas and as a channel to the larger user community.

In the final article, James Rush discusses the increasingly important area of library automation systems and networks. He points out that libraries have long been highly labor intensive and thus costly. Accordingly, since the early 1960s computer support for library operations has been looked upon as a major means of achieving increases in library productivity Over the last decade significant increases have been realized. Computers have been able to increase productivity as well as to provide new and improved library services. It is further stated by Rush that until the advent of automation the most significant previous technological innovation in libraries was the introduction of photocopiers. Rush's article focuses on the advances in applications of computers and related technologies to library operations that have been achieved in the last decade or so. He divides library operations into three major categories: (1) administration and management, (2) technical services, and (3) public services. He goes on to discuss operations in each of these areas as they are affected by the use of computer technology. He concludes with a discussion of the effects of technological advances on the future of library automation and networks. He points out that today only about 5% of the more than 90,000 libraries in North America now have any form of computer support, but he expects that the number of libraries using computers to increase their efficiency and effectiveness will increase steadily for many years to come. He suggests that these increases will result most of all from radical changes in philosophy and practice in librarianship. He goes on to review some of the technological advances that can be expected to affect library automation and networking. Rush believes that results beneficial to libraries as they are known today will not be durable because what is being automated is, or soon will be, largely outmoded by technological advances in other areas of information science. The profession must give serious consideration to the role of the library in an electronic society.

Finally, I am pleased to thank the contributors to this volume. They have given extensively of their time and effort and have made this an important and timely contribution to their profession. Despite the many calls upon their time and their busy schedules, they have made the effort to write substantial review and tutorial contributions in areas in which they are particularly knowledgeable and in which they have interest. It has been a substantial sacrifice on their part which is greatly appreciated. Because of their efforts, this volume achieves a high level of excellence and it will be of considerable value for many years to come. It has been a rewarding experience for me to edit this volume and to work with these authors.

MARSHALL C. YOVITS

Contents

The Web of Computing: Computer Technology as Social Organization

Rob Kling and Walt Scacchi

Computer Design and Description Languages

Subrata Dasgupta

Microcomputers: Applications, Problems, and Promise

Robert C. Gammill

Query Optimization in Distributed Data Base Systems

Giovanni Maria Sacco and S. Bing Yao

Computers in the World of Chemistry

Peter Lykos

Library Automation Systems and Networks

James E. Rush

The Web of Computing: Computer Technology as Social Organization

ROB KLING

Department of Information and Computer Science and
Public Policy Research Organization
University of California at Irvine
Irvine, California

WALT SCACCHI

Department of Computer Science
School of Engineering
University of Southern California
Los Angeles, California

ISBN 0-12-012121-2

1. Introduction

Most scholarly and professional examinations of the social and economic repercussions of new computing developments are based on a highly simplified conception of computing and social life. This conception focuses on certain explicit economic, physical, or information processing features of a technology. For the sake of simplicity, the social context in which the technology is developed and used, and the history of participating organizations, are ignored. This conception also assumes rationality on the part of computing developers and users which is neither very short term nor based on narrow interests. Furthermore, it sees the main repercussions of a new technology as direct translations of technical attributes into social attributes (e.g., faster data flows mean faster and better decisions). This conception informs many analyses of computing and related activities: of the payoffs of new computing developments, of the costs to be borne by developers and the broader public, of the extent to which different management strategies will yield more reliable and broadly workable computer-based systems. We call an analysis of computing which depends on these conceptions a *discrete-entity* analysis, and such a model of computing (activity) a discrete-entity model.

Examples of published analyses which rely upon a discrete-entity model of computing are very common, and they form the conventional logic for understanding the social repercussions of new computing developments. Most of the accounts emphasizing the benefits of new computer technologies rely on discrete-entity models, and criticisms of new technical developments can also be based on these models. For example, some analysts suggest that computerized systems will ensure that decision makers are aided by the selective ability of computers to sift through mounds of data, while other analysts argue that computers will rapidly increase the reams of data that decision makers must contend with. Despite these substantial differences in the conclusions reached, if the underlying model of computer use isolates it from the actual work practices and organization of labor within which automated data systems are typically developed and used, it is a discrete-entity analysis.

Careful empirical studies of the actual outcomes of using automated technologies such as automated information systems have often found that the expected direct benefits do not materialize easily (Kling, 1978; Laudon, 1974; Colton, 1978; Kraemer and King, 1981a). In special cases, some automated systems are simply dismal failures (Dery, 1977; Brewer, 1973); they are delayed in development or sometimes abandoned late in development, and if they work they are irrelevant or have to be worked around (Comptroller General, 1980). Analysts who use a discrete-entity model explain the occurrence of difficulties in realizing the direct benefits of new computing technologies and the occurrence of unexpected negative side effects or outright failures by attributing blame to a set of discretely identifiable causes: the operating managers were inadequate; the computer used was too small or cumbersome; promised data were not available; users were disinterested; there was insufficient support from top managers; etc.

We believe that when discrete-entity models are applied to analyses of the social consequences of socially complex computer applications and the difficulties encountered in their use, the results are often misleading. This article explains some of the limits of the conventional discrete-entity models and develops a richer family of models—*web models*—which we believe help make better predictions of the outcomes of using socially complex computing developments.

In contrast to the discrete-entity models, which gain simplicity by ignoring the social context of computing developments, web models make explicit the salient connections between a focal technology and its social and political contexts. In this article we indicate that many published analyses of computing developments are based on discrete-entity or web models. Unfortunately, most social analysts of computing (or other high technologies) rarely make their models explicit. We have abstracted the discrete-entity and web models from a wide variety of professional and scholarly studies.

The contrast between discrete-entity and web models can be usefully illustrated by a simple example drawn from another technology: automobiles. Suppose a traffic analyst is asked to predict the effects of building a new freeway connecting an urban center to a rural resort and passing through a suburban ring surrounding the city. If the analyst employs a discrete-entity model, he is likely to treat demand for trips as fixed, view the freeway as a medium for expanding travel capacity, and conclude that the new freeway will reduce congestion on existing highways. If he uses a web model, he will view the new freeway as part of a larger transportation system which is periodically congested. He is unlikely to simply assume that travel demand is fixed. Based on the history of other highway projects in which demand for short trips rather than total demand is fixed, he will consider the possibility that the new freeway will help satisfy demand for speedy trips. Net travel may increase. He may conclude that the new freeway will increase the volume of net travel between the suburbs and downtown, and between the metropolitan area and the rural resort, but that travel times will not be appreciably decreased during commuter rush hours and peak vacation times. In short, the planner who uses a web model is more likely to see a technical change (or new policy) as embedded in a larger system of activity, as having consequences which depend on peoples' actual behavior, and as taking place in a social world in which the history of related changes may influence the new change.

In this simple example the web model is an open systems model. But web models can be more far-reaching than simply being a version of "the systems approach." A politically oriented web analyst may further ask how the proposed freeway is routed and whose lands are crossed and whose avoided (Caro, 1975). He may also inquire into the interests which support the freeway and those (if any) which oppose it so as to further understand those interests it serves (Cornehls and Taebel, 1977, Chapter 5). Web analysts examine the interaction between people and technologies as part of a larger social and technical mosaic in which the development and use of the focal technology is embedded. In the case of technologies such as automobiles or electronic funds transfer systems, using the technology requires one to negotiate a complex set of institutionalized arrangements as well as deal with the equipment.

By making these models explicit, we hope to shed some light on critical assumptions which inform many published analyses of computing. The two models reveal important *emphases* of these analyses. Like any idealization, they sharpen important differences; but they do not exaggerate them. For example, no one is completely insensitive to the political context in which an organization develops a computerized model. Some analysts assume that the political relationships between an organization which uses computerized models and other groups in its environment will significantly influence the kind of model adopted, the way it is used, and the repercussions of that use (*web emphasis*) (Kling, 1978b); technical details of the model and its computer implementation will be of secondary importance. Other analysts hold that the adoption of a computerized mathematical model to replace manual or intuitive calculations has far greater influence on an organization's behavior with respect to computing than the political environment of the adopting agency (*discrete-entity emphasis*) (Simon, 1977). We have abstracted these two different emphases into the discrete-entity and web models.

While we have reason to favor web analyses, we believe that discrete-entity models are sometimes useful. Their simplicity makes them tractable. The analyst who employs a discrete-entity model need focus only on selected technical and economic characteristics of a new computing technology or application. In contrast, the analyst who uses a web model must work much harder. He must examine the social and political context in which the technology is developed and used, the going concerns of the organization using it, and the extent to which financial, technical, and staff resources support it. Sometimes these additional data yield important and surprising findings (Kling, 1978b; Danziger *et al.*, 1982), but this is an empirical matter.

Analysts who employ discrete-entity models mistakenly assume that they are universal in their application. Researchers who employ web models usually examine socially complex technologies, but they do not have sharp, simple descriptions of the situations in which they best apply. We shall explain their nature in this article, and we hope the reader will develop some good intuitions about these models. For now, we can say that they certainly apply to situations in which many organizations participate in the development, maintenance, and use of a computerized system. Large computerized systems such as air traffic control, multiservice military command-and-control systems, and multibank electronic funds transfer are all in this class. But so are many smaller systems which are developed for organizations by outside contractors or which are simply developed within the organization through the collaboration of three or more departments.

In this article, we explain the meaning of web models for understanding the dynamics of computing development and use in organizational life. We also examine the relative explanatory power of the discrete-entity model and web model by drawing upon the existing research literature and three case studies. The main question we ask about these models is not, Which is truest? Rather, we ask, What kinds of insights does each model give into the social dynamics of computing development and use? We also inquire about the ways in which each model provides analytical power for making evaluations and predictions. It will become apparent, after we examine these models more carefully in Section 2, that they conceptualize two theoretical extremes of a wide array of intermediate analyses of computing developments. Particular analyses are likely to be based on critical assumptions closest to the axioms of one or the other of these models.

Since web analyses examine the social and economic organization of computing activities, it is helpful to introduce some analytical terms useful for social analyses. Since these concepts are relatively abstract, we believe that a short case study can be helpful for giving the reader concrete illustrations of them. This little case study—of an automated inventory control system in the manufacturing firm Audiola—is introduced in Section 3. The key theoretical concepts are explained informally in Section 4, and are developed more formally in Appendix A. In Section 5, five major propositions that web analysts make about the dynamics of computing development and use are examined and explained by reference to the Audiola case. In Section 6 two additional cases of computer development and use are introduced. Section 7 examines these cases, and Section 8 concludes.

2. Models of Computing: Discrete-Entity and Web

The contrast between the discrete-entity and web models has recently been introduced into the computing literature under a different label: *tool model* versus *package model* (Conery, 1980; Kling and Dutton, 1982; Kling and Scacchi, 1979a,b, 1980; Sterling, 1979). For example, Kling and Scacchi (1979a, p. 108) characterize the tool model in which

> one can safely focus on the device to understand its use and operation. In contrast, the package metaphor describes a technology which is more than a physical device. ... the package includes not only hardware and software, but also a diverse set of skills, organizational units to supply and maintain computer-based services and data, and sets of beliefs about what computing is good for and how it may be used efficaciously. Many of the difficulties users face in exploiting computer-based systems lie in the way in which the technology is embedded in a complex set of social relationships.

In reviewing this new literature which examines the package metaphor, we have found it useful to sharpen this characterization. We have also found it useful to change the labels to "discrete-entity" and "web" to better reflect the differences between the two models. This section defines each of these two models by five properties.

Discrete-entity models of computing assume the following:

(a) A computing resource is best conceptualized as a particular piece of equipment, application, or technique which provides specifiable information processing capabilities. (*i*) Each computing resource has benefits, costs, and skill requirements which are largely identifiable. (*ii*) Computer-based technologies are tools, and are socially neutral.