

Computer Supported Co-operative Work

Edited by

Michel Beaudouin-Lafon

Université Paris-Sud, France

JOHN WILEY AND SONS

Chichester • New York • Weinheim • Brisbane • Singapore • Toronto

Copyright © 1999 by John Wiley & Sons Ltd,
Baffins Lane, Chichester
West Sussex PO19 1UD, England

National 01234 779777
International(+44) 1234 779777

e-mail (for orders and customer service enquiries): cs-books@wiley.co.uk

Visit our Home Page on <http://www.wiley.co.uk>
or
<http://www.wiley.com>

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except under the terms of the Copyright, Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency, 90 Tottenham Court Road, London, W1P 9HE, UK, without the permission in writing of the publisher.

Other Wiley editorial offices

John Wiley & Sons, Inc., 605 Third Avenue,
New York, NY 10158-0012, USA

WILEY-VCH GmbH, Pappelallee 3,
D-69469 Weinheim, Germany

Jacaranda Wiley Ltd, 33 Park Road, Milton,
Queensland 4064, Australia

John Wiley & Sons (Asia) Pte Ltd, 2 Clementi Loop #02-01,
Jin Xing Distripark, Singapore 129809

John Wiley & Sons (Canada) Ltd, 22 Worcester Road,
Rexdale, Ontario M9W 1L1, Canada

Library of Congress Cataloging-in-Publication Data

Beaudouin-Lafon, Michel.

Computer supported co-operative work / Michel Beaudouin-Lafon

p. c. — (Trends in software ; 7)

Includes bibliographical references and index.

ISBN 0-471-96736-X (alk. paper)

1. Teams in the workplace — Computer networks. I. Title

II. Series.

HD66.2.B4 1999

658.4'02 — dc21

98-42699

CIP

British Library Cataloguing in Publication Data

A catalogue record of this book is available from the British Library.

ISBN 0-471-96736-X

Produced from Postscript files supplied by the author.

Printed and bound in Great Britain by Biddles, Guildford, UK.

This book is printed on acid-free paper responsibly manufactured from sustainable forestry, in which at least two trees are planted for each tree used in paper production.

Contents

Series Editor's Preface	ix
Preface	xi
List of Authors	xv
1 Designing Groupware Applications: A Work-Centered Design Approach	1
<i>K. Ehrlich</i>	
1.1 Introduction	1
1.2 Requirements	4
1.3 Design	14
1.4 Deployment and Adoption of the Application	18
1.5 Case Study: TeamRoom	20
1.6 Summary	23
Acknowledgements	24
References	24
2 Workflow Technology	29
<i>C.A. Ellis</i>	
2.1 Overview	29
2.2 Workflow Concepts and Architecture	30
2.3 Historical Perspective and Related Work	36
2.4 Workflow Models and Modeling	38
2.5 Workflow Meta-Model	40
2.6 Example Systems	43
2.7 Research Directions and Issues	46
2.8 Summary	51
References	51

3 Media Spaces: Environments for Informal Multimedia Interaction	55
<i>W.E. Mackay</i>	
3.1 Introduction	55
3.2 Early Media Spaces	57
3.3 RAVE: EuroPARC's Media Space	58
3.4 Other Major Media Spaces	67
3.5 WAVE: A Detailed Case Study	71
3.6 Ethical Issues	77
3.7 Conclusion	79
Acknowledgements	79
References	79
4 Integration of Shared Workspace and Interpersonal Space for Remote Collaboration	83
<i>H. Ishii</i>	
4.1 Introduction	83
4.2 TeamWorkStation-1 and Seamless Shared Workspaces	86
4.3 TeamWorkStation-2 for N-ISDN	89
4.4 Seamless Integration of Interpersonal Space and Shared Workspace	92
4.5 Design of ClearBoard-1	93
4.6 Design of ClearBoard-2	95
4.7 Summary and Future Work	97
Acknowledgements	99
References	100
5 Group Editors	103
<i>A. Prakash</i>	
5.1 Introduction	103
5.2 Examples of Group Editors	104
5.3 Group Editor Architecture	107
5.4 Concurrency Control	109
5.5 Undo in a Group Editor	117
5.6 Supporting Collaboration Awareness	123
5.7 Design of Document Structure	126
5.8 Other Design Issues	127
5.9 Future Work	131
References	131
6 Groupware Toolkits for Synchronous Work	135
<i>S. Greenberg and M. Roseman</i>	
6.1 Introduction	135

6.2	Run-Time Architectures	136
6.3	Programming Abstractions	144
6.4	Groupware Widgets	150
6.5	Session Management	158
6.6	Conclusion	161
	Acknowledgements	164
	References	164

7 Architectures for Collaborative Applications 169

P. Dewan

7.1	Introduction	169
7.2	Collaboration Model	170
7.3	Generic Architecture	172
7.4	Design Space	174
7.5	External Modules	185
7.6	Rules	187
7.7	Classifying Existing Systems	188
7.8	Conclusions and Future Work	189
	Acknowledgements	191
	References	191

8 Software Infrastructures 195

P. Dourish

8.1	Introduction	195
8.2	Infrastructure Elements in CSCW	196
8.3	Communication	199
8.4	Coordination	202
8.5	Storage	205
8.6	Infrastructure and Specialization	207
8.7	Summary	216
	Acknowledgements	217
	References	217

9 Expanding the Role of Formal Methods in CSCW 221

C. Johnson

9.1	Introduction	221
9.2	Starting From the Ground Up: The Application of Formal Methods to CSCW	228
9.3	Dialogue Sequences	230
9.4	Formalizing the Presentation of CSCW Systems	236
9.5	Working Environments	243
9.6	Representing Workstation Layout	245
9.7	Using Ergonomic Guidelines to Inform CSCW Design	246
9.8	Prototyping	248
9.9	Conclusion	252

References

254

Index

257

Series Editor's Preface

During 1990, the twentieth anniversary of *Software Practice and Experience*, two special issues (one on UNIX Tools and the other on the X Window System) were published. Each issue contained a set of refereed papers related to a single topic; the issues appeared a short time (roughly nine months) after the authors were invited to submit them. The positive experience with the special issues resulted in *Trends in Software*, a fast turn-around serial that devotes each issue to a specific topic in the software field. As with the special issues of *SP&E*, each issue of *Trends* will be edited by an authority in the area.

By collecting together a comprehensive set of papers on a single topic, *Trends* makes it easy for readers to find a definitive overview of a given topic. By ensuring timely publication, *Trends* guarantees readers that the information presented captures the state of the art. The collection of papers will be of practical value to software designers, researchers, practitioners and users in that field.

Papers in each issue of *Trends* are solicited by a guest editor who is responsible for soliciting them and ensuring that the selected papers span the topic. The guest editor then subjects each paper to the rigorous peer review expected in any archival journal. As much as possible, electronic communication (e.g. electronic mail) is used as the primary means of communication between the series editor, members of the editorial board, guest editor, authors, and referees. A style document and macro package is available to reduce the turn-around time by enabling authors to submit papers in camera-ready form. Papers are exchanged electronically in an immediately printable format.

Trends will appear roughly twice a year. We now have issues in interactive data visualization techniques and computer supported cooperative work. Topics to be covered in forthcoming issues include other novel aspects of software.

The editorial board encourages readers to submit suggestions and comment. You may send them via electronic mail to **bala@research.att.com** or by postal mail to the address given below.

I would like to thank the editorial board as well as the staff at John Wiley for their help in making each issue of *Trends* a reality.

Balachander Krishnamurthy
Room D-229
AT&T Labs—Research
180 Park Avenue
Florham Park NJ 07932
USA

Preface

Computer Supported Cooperative Work, or *CSCW*, is a rapidly growing multi-disciplinary field. As personal workstations get more powerful and as networks get faster and wider, the stage seems to be set for using computers not only to help accomplish our everyday, personal tasks but also to help us communicate and work with others. Indeed, group activities occupy a large amount of our time: meetings, telephone calls, mail (electronic or not), but also informal encounters in corridors, coordination with secretaries, team workers or managers, etc. In fact, work is so much group work that it is surprising to see how poorly computer systems support group activities. For example, many documents (such as this book) are created by multiple authors but yet no commercial tool currently allows a group of authors to create such shared documents as easily as one can create a single-author document. We have all experienced the nightmares of multiple copies being edited in parallel, format conversion, mail and file transfers, etc.

CSCW is not recent. Back in the late 1960s, Doug Engelbart created the NLS/Augment system that featured most of the functions that today's systems are trying to implement such as real-time shared editing of outlines, shared annotations of documents, and video-conferencing. The field really emerged in the 1980s and has been growing since then, boosted in the recent years by the explosion of the Internet and the World Wide Web. The Web itself is not a very collaborative system: pages can be easily *published* but it is impossible (or very difficult) to *share* them, e.g. to know when someone is reading a particular page or when a page has been modified. The range and complexity of the problems to solve to support cooperative activities is rapidly overwhelming: data sharing, concurrency control, conflict management, access control, performance, reliability, the list goes on.

A large part of this book is devoted to the exploration of these problems and the state of the art of their solutions. In fact, *CSCW* is challenging most of the assumptions that were explicitly or implicitly embodied in the design of our current computer systems. *CSCW* tools, or *groupware*, are by nature distributed and interactive. To succeed in the marketplace, they must be safe (authentication), interoperable (from network protocols to operating systems and GUI platforms), fault-tolerant and robust (you don't want to be slowed down or lose your data if another participant in the session uses a slow connection or experiences a crash).

In addition to these technical difficulties, there is another, maybe harder, problem in implementing groupware: people. For a medium to work, there must be an audience that accepts using it. Usability issues have stressed the need to take the users into account when designing, developing and evaluating an interactive software. For groupware, usability issues go beyond the now well-understood (if not always well-applied) methods from psychology and design. They involve social sciences to understand how people work together, how an organization imposes and/or adapts to the work practices of its workers, etc. In many *CSCW* projects,

ethnographic studies have been conducted to better understand the nature of the problem and the possible solutions. A large body of the research work in CSCW is conducted by social scientists, often within multidisciplinary teams. Computer scientists often ignore or look down upon this aspect of CSCW and almost always misunderstand it. User-centered design is essential to ensure that computer scientists solve the right problems in the right way. Traditional software works as soon as it “does the job”; Interactive software works better if it is easy to use rather than if it has more functions; Groupware works only if it is compatible with the work practices of its users.

Overview of the book

This book attempts to cover the broad field of CSCW and to give an overview of the history, state of the art and research issues of this exciting field. It is divided into two parts: the first part covers *groupware tools* while the second part covers *tools for groupware*.

The first chapter by Ehrlich focuses on a category of groupware for asynchronous group work such as the well-known Lotus Notes. More importantly, it provides an in-depth analysis and a set of recommendations to help design, develop and deploy groupware in an organization. Ehrlich emphasizes that groupware is for group work and therefore all aspects of group work must be well understood for the software to be accepted.

Chapter 2 by Ellis covers workflow systems. Since the 1960s, businesses have been converting their manual or mechanical information systems into computerized systems. Workflow systems go beyond traditional information systems by embodying a description of the work processes of the organization. The system therefore can be proactive, e.g. by automatically circulating documents or by reminding users of their duties when they are late. Ellis analyzes the promises, realities and problems of this category of groupware.

Chapter 3 by Mackay describes media spaces, i.e. communication systems that combine audio, video and computers to provide distant users with a means for social interaction and informal communication. Unlike videoconferencing rooms which require reservations and inevitably lead to formal meetings, media spaces attempt to broaden the bandwidth among users in order to support “real-life” human communication. Mackay covers the underlying design rationale of the existing systems and raises awareness on ethical and privacy issues of groupware.

Chapter 4 by Ishii describes systems that allow small groups to work in a tightly-coupled way at a distance, such as an instructor and a student or a group of designers. The chapter is illustrated by a description of a series of prototypes developed by Ishii and his group. While the prototypes are technically more and more complex, the chapter shows how the observation of the type of group work that was to be supported leads from one prototype to the next.

Chapter 5 by Prakash covers shared editors, editors that can be used by several users simultaneously to edit, in real-time, a single document. It marks the division between the two parts of the book: the concepts of shared editor are introduced and some examples are presented. The chapter then goes into an in-depth description of the techniques used to implement shared editors, focusing on issues such as managing the consistency between several copies of the document being edited and implementing multi-user undo.

Chapter 6 by Greenberg and Roseman describes groupware toolkits. In the same spirit as user interface toolkits, groupware toolkits provide programmers with predefined components that help implement groupware tools. The chapter covers toolkits for real-time (or synchronous) groupware, with components such as group widgets, awareness widgets, session

managers, etc. Greenberg and Roseman use their own toolkit, GroupKit, to illustrate the design issues of such tools.

Chapter 7 by Dewan covers software architectures for CSCW. Since groupware applications must interact, by definition, with several users, they are in general distributed over a network. Dewan systematically examines the various ways in which an application can be decomposed into modules, threads and processes and the many tradeoffs that the various solutions incur. This leads to a set of measures for an architecture that help better understand this large design space.

Chapter 8 by Dourish covers software infrastructures, i.e. the types of services that are or could be provided by the operating system, network and other middleware to implement groupware applications. Given the varying needs of groupware applications, Dourish presents a particular approach, open implementation, as particularly promising since it combines flexibility, performance and openness.

Chapter 9 by Johnson provides an original perspective on the role of formal methods in CSCW, more particularly in the requirements phase of development. Johnson introduces several formal notations and models and uses examples to show how they can be applied to practical cases.

CSCW radically changes the status of the computer. Until now, the computer has been used as a *tool* to solve problems. With CSCW, the computer/network is a *medium*: a means to communicate with other human beings, a vector for information rather than a box that stores and crunches data. If we look at the history of technology, new media have been much more difficult to invent, create and operate than new tools. From this perspective, it is not surprising that CSCW has not yet realized its full potential, even in the research community. I hope this book will help readers to better understand the challenges and promises of CSCW and encourage new developments both in research and in industry.

Michel Beaudouin-Lafon

Laboratoire de Recherche en Informatique

Bâtiment 490

Université de Paris-Sud.

91 405 Orsay Cedex FRANCE

mbl@lri.fr

List of Authors

Prasun Dewan

Department of Computer Science
University of North Carolina
Chapel Hill, NC 27599
USA
dewan@cs.unc.edu

Kate Ehrlich

Lotus Development Corp.
55 Cambridge Parkway
Cambridge MA 02142
USA
kate.ehrlich@lotus.com

Saul Greenberg

Department of Computer Science
University of Calgary
Calgary, Alta T2N 1N4
CANADA
saul@cpsc.ucalgary.ca

Chris Johnson

Glasgow Interactive Systems Group
Department of Computer Science
University of Glasgow
Glasgow G12 8QQ
SCOTLAND
johnson@dcs.gla.ac.uk

Atul Prakash

Department of EECS
University of Michigan
Ann Arbor, MI 48109-2122
USA
aprakash@eecs.umich.edu

Paul Dourish

Xerox PARC
3333 Coyote Hill Road
Palo Alto, CA 94304
USA
dourish@parc.xerox.com

Clarence Ellis

Department of Computer Science
University of Colorado
Boulder, CO 90309-0430
USA
skip@cs.colorado.edu

Hiroshi Ishii

MIT Media Laboratory
20 Ames Street
Cambridge, MA 02139
USA
ishii@media.mit.edu

Wendy E. Mackay

Department of Computer Science
Aarhus University
Aabogade 34
DK-8200 Aarhus N
DENMARK
mackay@daimi.aau.dk

Mark Roseman

Department of Computer Science
University of Calgary
Calgary, Alta T2N 1N4
CANADA
roseman@cpsc.ucalgary.ca

