TRUST IN TECHNOLOGY:
A SOCIO-TECHNICAL PERSPECTIVE

Computer Supported Cooperative Work

Volume 36

Trust in Technology: A Socio-Technical Perspective

Edited by

Karen Clarke
Lancaster University, United Kingdom

Gillian Hardstone
Edinburgh University, United Kingdom

Mark Rouncefield
Lancaster University, United Kingdom

and

Ian Sommerville
Lancaster University, United Kingdom

Springer

A C.I.P. Catalogue record for this book is available from the Library of Congress.

ISBN-10 1-4020-4257-4 (HB)
ISBN-13 978-1-4020-4257-7 (HB)
ISBN-10 1-4020-4258-2 (e-book)
ISBN-13 978-1-4020-4258-4 (e-book)

Published by Springer,
P.O. Box 17, 3300 AA Dordrecht, The Netherlands.

www.springer.com

Printed on acid-free paper

Printed in the Netherlands.

CONTENTS

List of Contributors vii

Introduction: A new Perspective on the Dependability of Software Systems - Graham Button ix

Chapter 1: Trust and Organisational Work - Karen Clarke, Gillian Hardstone, Mark Hartswood, Rob Procter, Mark Rouncefield 1

Chapter 2: When a Bed is not a Bed: Calculation and Calculability in Complex Organisational Settings -Karen Clarke, John Hughes, Mark Rouncefield, Terry Hemmings 21

Chapter 3: Enterprise Modeling based on Responsibility - John Dobson and David Martin 39

Chapter 4: Standardization, Trust and Dependability - Gillian Hardstone, Luciana d'Adderio and Robin Williams 69

Chapter 5: 'Its About Time': Temporal Features of Dependability - Karen Clarke, John Hughes, Dave Martin, Mark Rouncefield, Ian Sommerville, Alexander Voß, Rob Procter, Roger Slack and Mark Hartswood..105

Chapter 6: Explicating Failure - Karen Clarke, Dave Martin, Mark Rouncefield, Ian Sommerville, Alexander Voß, Corin Gurr, Rob Procter, Roger Slack, Mark Hartswood 123

Chapter 7: Patterns for Dependable Design - David Martin, Mark Rouncefield and Ian Sommerville 147

Chapter 8: Dependability and Trust in Organisational and Domestic Computer Systems - Ian Sommerville, Guy Dewsbury, Karen Clarke, Mark Rouncefield 169

Chapter 9: Understanding and Supporting Dependability as Ordinary Action - Alexander Voß, Rob Procter, Roger Slack, Mark Hartswood and Mark Rouncefield 195

Chapter 10: The DIRC Project as the Context of this Book - Cliff B Jones 217

LIST OF CONTRIBUTORS

Graham Button - Xerox Research Centre Europe, Grenoble France
Karen Clarke - Department of Computing, Lancaster University, UK
Luciana d'Adderio - Research Centre for Social Sciences, University of Edinburgh, UK
Guy Dewsbury - Department of Computing, Lancaster University, UK
John Dobson - Department of Computing, Lancaster University, UK
Corin Gurr - Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK
Gillian Hardstone - Research Centre for Social Sciences, University of Edinburgh, UK
Mark Hartswood - Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK
Terry Hemmings - Department of Computing, Nottingham University, UK
John Hughes - Department of Sociology, Lancaster University, UK
David Martin - Department of Computing, Lancaster University, UK
Rob Procter - Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK
Mark Rouncefield - Department of Computing, Lancaster University, UK
Roger Slack - Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK
Ian Sommerville - Department of Computing, Lancaster University, UK
Alexander Voß - Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK
Robin Williams - Research Centre for Social Sciences, University of Edinburgh, UK

INTRODUCTION: A NEW PERSPECTIVE ON THE DEPENDABILITY OF SOFTWARE SYSTEMS

Graham Button
Xerox Research Centre Europe, Grenoble France

1. INTRODUCTION

The Chapters that make up this book all report on research that has been conducted within two strands of the Interdisciplinary Research Collaboration in Dependability (DIRC) project[1] sponsored by the Engineering and the Physical Sciences Research Council (EPSRC)[2]. The majority of the research on which this book is based has been conducted within the research strand entitled, 'Organizational Culture and Trust', with in-put from the 'Human Interaction in Real-Time Systems' strand of research. As the titles of these strands suggests, this book, and indeed the whole DIRC project, does not consider the matter of the dependability of software systems in the traditional terms and methodologies of software engineering and computer science.

Questions relating to the dependability of software systems have, in the main, been asked with respect to control and safety critical systems, iconic examples of these being systems deployed within the nuclear power industry, air traffic control, and carrier flight-deck operations. They have also traditionally been considered within software engineering and computer science in terms of formal metrics that model and measure the tolerances of systems. The DIRC programme, however, is providing a radical overhaul of

[1] DIRC includes researchers from five British Universities established in the area of dependable computer systems and related topics: City University, Lancaster University, The University of Edinburgh, The University of Newcastle and The University of York. DIRC aims to develop knowledge, methods and tools to ensure more dependable computer-based systems. The project started in September 2000, and is now half-way through its lifetime.

[2] The Engineering and Physical Sciences Research Council (EPSRC) is the UK Government's leading funding agency for research and training in engineering and the physical sciences.

the way in which the dependability of software systems is approached and assessed. It is radical in the sense that it is moving the issue beyond the confines of the disciplinary boarders of software engineering and computer science. It is doing so by bringing into consideration the cultural, organizational, interactional and psychological context in which systems are used, and also through the way in which dependability can be assessed through statistical methods. This radical overhaul brings together a multidisciplinary team of researchers drawn from the disciplines of statistics, sociology and psychology, in addition to computer science and software engineering.

DIRC is a seven year program, which began in September 2000. As evidenced by this collection, it has so far developed two foundationally innovative ways through which the dependability of software systems can be considered and which have set the tone for the research results that have so far been published.[3] The first is that the project has made dependability an issue that is relevant for software systems in the work place, *per se*, not just systems that are used in control and safety critical environments. This is an important step because even though the safety of an environment may not be at issue, nevertheless the dependence of an organization on, for example, its accountancy system may be crucial for its economic survival. Encouraging designers to view the dependability of any workplace system as at least a latent requirement emphasizes the need for design to understand the organizational context within which systems are placed. In this respect all of the chapters in this book investigate the issue of dependability outside of the confines of control and safety critical environments. For example, Clarke, Hughes and Rouncefield's study in Chapter 2 is situated within a large hospital trust in the North of England; Voß, Proctor, Slack and Hartswood's Chapter 9 investigation examines work in manufacturer of mass-customized diesel engines; Dewsbury, Sommerville, Clarke and Rouncefield in Chapter 8 move into the domestic setting, Rouncefield et al in Chapter 5 draws off work conducted in the diverse settings of a hospital, a steel-works, and an engine manufacturer, while Hardstone, D'Adderio and Williams in Chapter 4 is based on studies conducted in an automobile manufacturer, a high-end electronics company and another NHS hospital trust.

[3] DIRC has assembled an impressive list of publications to date which can be accessed at: www.dirc.org.uk/publications/index.php

The second innovation is to have extended the type of question that is relevant with respect to the issue of dependability. This has itself been done in two ways. First, not only has DIRC extended the range of systems to which questions of dependability can be addressed, as discussed above, it has also extended the whole idea of what a system is. Thus DIRC considers a system not just in terms of the computational technology involved, in DIRC terms the system includes the user and the organizational arrangements of use as well as the actual technology. This can be seen particularly in parts of Hardstone et al's Chapter 4 where they contrast two organizations using the same technology but with very different levels of reliability. It is clear from this chapter that assessing the dependability of the system inevitably has to involve reference to the organizational circumstances of the use of technology and that the dependability of the technology is a consequence of those circumstances. The upshot of expanding the idea of a system beyond the computational technology involved to also include users and the organizational environment and culture milieu within which the technology is deployed and used is that even if the computational technology itself is measured to be fit for its purpose according to formal models of dependability, problems with the usability, and acceptability of the technology, or the possibility of human error, bring into question the dependability of the system *as a whole*. Second, DIRC has not only extended the concept of 'system' it has also provided additional criteria to those of performance metrics for modeling the dependability of systems. Thus in Chapter 8 Dewsbury, Sommerville, Clarke and Rouncefield extend technically based models of dependability. They argue that Laprie's model needs to be expanded to include matters such as user acceptability and the ability of a system to be able to adapt to different environments of use and different users.

2. MAKING DEPENDABILITY AN EVERYDAY MATTER

This collection brings together research that represents the way in which DIRC researchers have accomplished an overhaul of the concept of the dependability of software systems through the examination of the social and organizational issues surrounding dependability. The idea of 'trust' that figures in the title is being used to gather together this new way of thinking about dependability as a culturally and organizational embedded matter. Dependability is a concept that has been appropriated by the engineering disciplines and given a technical meaning defined in terms of performance metrics and tolerances. Reapplied to the everyday world that technical definition of dependability would not be recognizable because in the everyday world dependability means much more than is meant by the technical definition. Yet, as DIRC research is showing, dependability is very much an everyday, not just a technical matter. The utilization of the concept of 'trust' is a way of freeing ourselves from the technical definition of dependability by making dependability a more everyday matter and thus freeing up the ideas involved from their technical specification. For example, systems that patently return false information, may, technically speaking in terms of their design parameters, be judged not to be dependable. However, as Clarke et al show in Chapter 2, in the everyday world of work, systems that may be formally defined in technical terms not to be dependable, may, nevertheless, be perfectly usable for the practical purposes of accomplishing the work they support. In these terms, although they may not be formally dependable they may be, nevertheless, 'trusted' by their users and within the organization. Within the everyday world, as Voß et al demonstrate in Chapter 9 they may be dependable enough for the practical purposes of getting the job done within the practical circumstances in which they are used.

Computer systems populate the everyday world. Systems pervade the everyday world of the work place: document systems, work flow systems, control systems, accountancy systems, and so on. The everyday world of the home is integrated into organizational systems: we use the phone, we receive bills generated by computer systems, and many people also explicitly use computer systems within the home. Systems exist within our everyday world of transport, holidays, leisure, medicine and the rest. Computer systems are just, quite simply, an everyday matter. And *trusting* a technology is also an everyday matter. That is, the determination of whether a system is trusted or not consists in the ways in which people make everyday judgments about trust which may accord, but which may also not accord with criteria that designers and engineers use to determine the dependability of a system. In this respect the question of how, in the everyday world, people determine that they trust something or someone is appropriate for understanding how people put, or do not put, their trust in a system.

So what does trust in the everyday world consist of? One of the points that Harold Garfinkel (1963) made in a paper on experiments with trust, is that trust is a background, taken for granted, expectancy in everyday interaction. So, for example, we take for granted that people will understand what we mean within a particular context: when we walk into the newspaper shop in the UK and ask for The Guardian we never think that instead of being handed a newspaper by that name we might be offered a person dressed up in a Superman like costume. We take for granted that when we tell someone to follow the signs to London they will follow the signs in the direction of the arrow head and not in the direction of the tail. It is just an ingredient of the way in which we conduct our everyday lives that there is a background expectancy that when we interact with people they have a commonsense knowledge of the social structure which they use to understand what they and others are saying and doing. To make this feature of social life explicitly visible Garfinkel would make trouble, and he describes how people become agitated when these background expectancies are breached (Garfinkel 1967) So, for example, he describes how he asked his students to continually question what members of their family were saying when they were watching television, and how quickly interactions deteriorated when they were constantly asked what they meant when they said something. In what they do and how they do it, people display that they just trust that they will be understood, unless understanding is marked as a particular, local problem. That trust comes to light when it is systematically breached.

In this respect, one of the ways in which we treat the issue of trust in the everyday world is that it is an *occasioned* matter. That is, we do not go around systematically asking if we trust or do not trust someone or something. Trust is a routine background expectancy in everyday interaction and in our everyday use of technology. To question this implicitly displayed trust is the result of something having occurred that brings it into question; it is a matter that has been brought about or occasioned by particular circumstances. For example, once discovered, the errant spouse may never be trusted again, or once someone misses an appointment we may wonder about their likelihood to do so again. We also do not go around our world making judgments that someone is trusting. We may make that observation but it is, again, occasioned by the circumstances. So we might say that the rock-climber standing on the crumbling ledge put his trust in his friend if we see him grasping for his fiends outstretch arm, but it is that occasion that makes relevant our observation about trust, for trust might not be part of the behaviour of just stretching out an arm. It is that action in the circumstances. We might also say that it takes time to build up trust in a new acquaintance or new things: we may have had doubts to begin with but we come to trust through experience that the new person in the office can be trusted to get the job done on time. Again, it is the circumstances, the fact of being new that occasions our question, once the new becomes familiar, the relevancy of the activity of trust disappears.

Computer systems are placed within the everyday world in which commonsense understandings of trust prevail, not into a world in which technical definitions of dependability rule. Like anything in that everyday world, they are therefore subject to commonsense everyday judgments with respect to their trustworthiness or their dependability. Thus, for instance, we may have had doubts about the new document repository system, but we come to trust, through experience, that it can be used to get the job done; as we near the end of the final quarter and the reporting deadline looms into view for tomorrow we might say of someone whom we observe setting off the long report to print overnight so that it is ready for the next day that they obviously trust the technology. We can also say that trust in technology is an occasioned matter. We regularly enter lifts without a moment's thought, but if the lift judders or stops short of a floor we may eye it suspiciously the next time we take it. In the everyday world we regularly use systems and our judgments as to their trustworthiness will be based upon our experiences; do those experiences with the technology occasion us to question their trustworthiness, or knowing their shortfalls do we trust them enough for them to be useful. We may have many occasions to wonder about the trustworthiness of systems per se, we read about large scale disasters involving air traffic control systems that result in the closure of airports, the UK stock exchange system that had to be scrapped, and the London ambulance fiasco. In is abroad in the world that the failure of computer systems can result in large amounts of wasted money and sometimes death. These reports may occasion us to be jaundiced about computer systems in general, yet our local experience of the system we use might mean that we log on at the beginning of the day without a moments thought.

All of the papers in this collection are about placing the issue of dependability on an everyday footing through the articulation of how, in the everyday world, people orient to the question of the trustworthiness of a system. They are also about making this everyday orientation a consideration in the design process. There are three themes that can be discerned that run through the chapters in this respect. First, trust as a practical matter in the situated affairs of users, the workplace and organizations. Second, the consequences of this for traditional software engineering and computer science understandings of dependability. Third, the utilization of ethnographic investigation as a design methodology.

3. DEPENDABILITY AS A PRACTICAL MATTER

There have been many ethnographically grounded studies of technology in use that describe the way in which technology in the workplace is made to work by those who use it, (cf. Luff et al 2000; Heath and Button 2002). Together, these studies have illustrated a number of themes with regard to the way in which people work to make technology usable in the workplace.

For example, it has been described how technology may be used in different ways or for different purposes to those intended by its designers (Bowers, et al 1995). Also, many studies have described how people have to work around a system in order to get the work done that the system is intended to support or automate. Quite simply, studies of technology in use have demonstrated that evaluating the role, efficiency, dependability and productivity gains of computer systems in the work place are complex matters which cannot be adequately appreciated through solely asking technical questions to do with performance, functionality, reliability, maitence and the like. The workplace is not just a space within which varieties of technologies are functioning according to their specifications and tolerances, it is a complex social milieu made up of matters of organizational and interactional contingencies which play into the very working, operation and assessment of technology. The technology found in the workplace does not stand outside of this social milieu, it is embedded within it, and it is no surprise, and in itself, no news, that technology is marbled through with social relevancies and social concerns.

Seen from a purely formal point of view the dependability and trustworthiness of a system may appear to be, in principal, the relatively straightforward matter of setting performance tolerances. This is not to say that it is straightforward to implement, indeed, it might involve the solution of difficult engineering problems but, in technical terms, a system either performs within the specified parameters or it does not. However, once a system is placed within the workplace it is placed within a social milieu within which organizational and interactional matters play upon how technology is considered. In the workaday world dependability may not then be *measured* but rather *judged*.

This idea is at the heart of Chapter 9 by Voß et al. They make the point that the dependability of a system is achieved through the actions and interactions of those involved in using the system. In this respect it takes interactional work by those involved to make a system dependable, for practical purposes. The starting point for this argument is to remember that 'dependability' is an everyday natural language expression used and understood in the daily round of everyday (including working) life. Voß et al thus want to understand what dependability means in everyday use within EngineCo a company that produces mass-customized diesel engines, and they make visible the ways in which, in their actions and interactions, the people involved at EngineCo come to view systems as dependable. They make visible the, borrowing a term from Livingstone (1986), 'lived work' that participants engage in, in order to make the systems they work with 'more or less dependable'. Thus for example, they describe the assembly control host which controls all processes within EngineCo and which interfaces with the company's EPR system (SAP R3) and with systems specific to the different functional units making up the plant. The idea of the 'buildability' of an engine might be thought of something that is verifiable by the system. That is, the buildability of an engine is dependent upon, for example, the necessary components being available. Thus the system can confirm or not that an engine can be built if the parts are available or not. However, there are a variety of contingencies that play into building engines, one of these is the short supply of some parts, and another is high customer demand. The buildability of engines is not then just dependent upon the availability of parts; work on building the engine may begin even if the parts are not available in order to work to meet demand. Work on the engine will be progressed until the absence of the part prevents them from continuing any further; but once the part arrives it can be finished off.

Working in this way means that more engines can be shipped to meet demand even though parts are missing at the time of starting to build the engine. Working in this way, though, requires co-ordination between different departments and Voß et al describe how the control room workers have to take account of the interests of the assembly workers in how they pace the flow of different types of engines to the assembly workers in order to have an even flow of complicated and less complicated assemblies that allow the performance targets to be achieved. The buildability of an engine is not then provided for in the verification by the system of the necessary components being available. The system may confirm or not confirm that it is possible. The buildability of an engine is an organizational judgment, which once made then requires organizational and interactional work to accomplish in such a way as to be done under the auspices of organizational constraints. The dependability of the system to verify or not the buildability of an engine is really irrelevant to those concerned; they work to make the system work for their situated purposes. Voß et al are thus making dependability a 'members' phenomenon' and argue that the professional understanding of dependability needs to be complemented with the practical view point of what dependability means for those who use systems.

One of the points raised in this chapter relates to an idea first introduced by David Sudnow (1993) in his account of the work of hospital staff: 'normal troubles'. Normal trouble are problems which arise in the course of work but which are just part and parcel of the work being done, and with which the people involved are familiar and which they have contended with and overcome on many occasions. The issue of normal troubles is relevant to how the dependability of a system is considered because systems can give rise to normal troubles in working routines. In this respect it might be supposed that a system is not performing properly, may not be dependable or trustworthy. However, the troubles that systems occasion may be very familiar and these troubles may be regularly and routinely handled with in the course of the work. Thus, although giving rise to problems a system may be deemed to be dependable and trustworthy for the practical purposed of doing the job. This is an idea that is articulated in Chapter 2 by Clarke et al.

They describe a ‘bed crisis’ inside a UK hospital trust. The management information system was alerting the Directorate Manager of Orthopaedics (DMO) that the hospital was ‘minus nine beds’. That is to say, that the hospital trust, which was made up of three hospitals, would not have enough beds to cope with the number of patients they expected they would need to care for. There would be a shortfall of nine beds that was deemed to be a crisis level caused by a traffic accident. Clarke et al describe, however, that the DMO worked to normalize this crisis: “we go through our usual rituals for situations like these”. These “usual rituals” turned out to be walking around the hospital wards, physically locating spare beds, inspected the ‘bed boards’ – physical representations of bed availability- asking ward sisters about their bed availability descriptions (data which is fed into the information system) and what they really meant. At the end of this ‘hands on process’ it was possible for the DMO to establish that rather than being nine beds short the hospital could in fact cope with the ‘crisis’. The problem was handled in the routine of the practices through which the DMO made her calculations. Clarke et el describe how the apparent solidity and objectivity of managerial information (as proposed by the system involved) can be continually challenged in the activities of those involved in establishing how many free beds there are in practice. Clarke et al thus elaborate the theme on the work of using the systems ‘the work of managing the bed management system, of making a system of calculability work’. However, they also elaborate how the fact that it requires work to make the system work, and that plainly the system did not reflect the real state of affairs on the wards does not mean that the system was viewed as not dependable, or trustworthy. It was just considered that this work of walking the wards and ascertaining the ‘true’ picture was part of using the system. Indeed, the system was very much viewed by those concerned as a support for their work, because it gave them, for their practical purposes, enough information to work with. The system, as Voß et al described with regard to their study, was more or less dependable for the practical affairs it was used, in part, to manage. Dependability viewed only in terms of a set of performance criteria would fail to capture the ways in which these performance criteria are both constituted and judged in the course of working with the system.

The issue of whether or not there are enough beds is, of course, a temporally located determination. Having, for example, five spare beds may be enough on this occasion, *this time*, but may not be enough on another occasion, at *that time*. In Chapter 5 Clarke et al consider the way in which *time* has been used to understand cultural aspects of technology and propose that 'timeliness' is a consideration in understanding how dependable systems are.

The social sciences have considered 'time' from the point of view of a number of social theories ranging from a Marxist interest in the regulation of time under modern capitalism to Giddens' concern with temporal change. In the vein of the studies presented in this collection, Clarke et al, however, eschew a theorizing approach to time and rather turn to the way in which time is actually oriented to in the actions and interactions of people working in organizations. By understanding the specifics of the way in which time is woven into organizational culture the authors intend to influence the design of systems whose dependability is, in part, measured temporally. Technologies which support 'just-in-time' manufacturing or IT systems which promise productivity improvements by enabling information to be organized and accessed in a more timely fashion cannot be assessed and judged just on an abstract and generalized measurement systems for time. Comparisons with a current state and a past state that show that it now takes, as measured in terms of speed, less time to perform an operation than it did before, miss the point of the way in which time may be calculated in organizational life. In this respect the dependability of a system that it will deliver in time or that it can be relied upon to be faster than its predecessor may not so much turn on a precise mathematical calculation but rather on the way in which time is accounted for and measured within the organizational context in which a system is to operate. To this end Clarke et al range across a variety of studies of the workplace to bring out how time is oriented to and impacts the organizational structure and work done within in it that a system will support. With those understandings in hand designers have a more sensitive appreciation of the situational relevance of time, and how to accommodate it in their design, than they would otherwise have through a purely generalized measurement system for time that stands outside of the situation in which 'the clock ticks'.

4. IMPLICATIONS FOR FORMAL MODELS OF DEPENDABILITY

The second theme that permeates the book concerns the implications that this work has for the formal conceptions of dependability found in software engineering and computer science. Most of the chapters invite systems' designers and builders to consider elaborating on the formal characterization of dependability through a consideration of the practical character of dependability and trustworthiness displayed in situated judgments. The Chapters 8,7 and 3, by Dewsbury, Sommerville, Clarke and Rouncefield; Martin and Sommerville, and by Dobson, however, explicitly address this issue. In Chapter 4, Hardstone, D'Adderio and Williams bridge between both the theme of dependability as a practical matter and that of the implications this has for formal methods of dependability, and in that capacity I will introduce this chapter first.

Hardstone et al consider the way in which formalists have approached dependability through the *standardization* of information structures and organizational practice. This is particularly relevant with respect to systems and practices that are operated across geographically dispersed sites where the need for coordination is important. Standardization is seen to be a way in which sites can come to trust each other's operations and reciprocal inputs. Hardstone et al, through a study of three organizations that were moving towards standardization, suggest, however, that standardization is really a practical matter, and more negotiable than is suggest by formalist approaches. As Hardstone et al explain, formalist accounts of standardization and classification in system design emphasize that ensuring consistency, completeness and mutually exclusive categories of classification will result in systems that are both usable and dependable. In reality, however, looseness in the system and trade-offs are required to make the standardization process workable. Hardstone et al thus draw conclusions about how formalists and information systems designers should approach the issue of standardization.

The three case studies that are presented represent different levels of organizational heterogeneity and diversity. ComputerCo, a manufacturer of high-end electronics, was attempting to standardize a product and its production processes across two geographically separated sites; MotorCo, an automotive manufacturer was trying to bring about standardization within a single organization, while NHS Urban, a UK National Health Service Trust hospital was introducing standardization of recording clinical practice across different professional bodies of healthcare practioners. In the case of ComputerCo the heterogeneity they were attempting to handle through standardization was differences in culture and labour structures, while within MotorCo it was the different ways in which two engineering groups who were cooperating in the production of a product worked and how they used different database languages, and within NHS Urban it was the fact that the different bodies of professionals had their own and different bodies of knowledge and practice. In ComputerCo the move towards standardization involved creating and implementing rules and methodologies to ensure that the product and processes to produce the product were exactly duplicated at each site; within MotorCo it involved the introduction of new software supported product structure and a single database, and within NHS Urban it consisted of introducing a computer based records system.

Hardstone et al acknowledge that all three organizations partially succeeded in their standardization attempts; however, what is of interest here is that they also partially failed because in the process of standardization new forms of undependability developed. In the case of ComputerCo not all lower level knowledge could be codified and transformed, within MotorCo, contrary to the arguments of codification economists, standardization did not resolve the existing incompatibilities between the sites and led to new bottlenecks, and for NHS Urban the difficulties of fitting the system to some of the activities of the different groups and the difficulties of negotiating a common ground between them emphasized how mediation between the groups had been necessary to make the data dependable. This was now difficult to accomplish and the consequence was that the data was undependable. The lesson that Hardstone et al draw for formalist approaches to dependability and for designers of information systems is that local meaning and practice are important and that coordinating systems need to be flexible enough to handle necessary local variations. A conclusion that echoes, and is thus reinforced, by arguments previously made about workflow systems (Suchman 1994).

In Chapter 8 Dewsbury, Sommerville, Clarke and Rouncefield focus on Laprie's dependability model. They argue that work on dependability has been mainly concerned with control and production systems. However, in the spirit of DIRC research they argue that with the proliferation of computer systems in the work place it is not just production and control systems that are critical; business and governments are in some cases totally reliant upon a system for crucial aspects of their operation, and the dependability of such systems is as critical as it is for the systems traditionally driven by dependability requirements. However, it is not only within the work place that dependability is of concern; some domestic-based computer systems, in the instance with which Dewsbury et al are concerned, assistive technology systems for older people, must also be dependable for they can involve life critical matters. A consequential feature of such domestic based systems is the extent to which the system is *acceptable* to a user and how well the system can adapt to different users and user environments. Thus the installation of domestic-based systems should not just be concerned with the need to make the system failure free, instead the overall dependability of a system also involves the issue of whether or not it fulfils its intended purpose from the user's point of view. Dewsbury et al make the point that technically based dependability models such as Laprie's do take account of users. However, users should be considered as elements in the system that are comparable with other elements such as hardware and software elements. In this respect 'interaction faults' can be seen as resulting from 'human errors' just as they can be considered as resulting from hardware and software. Dewsbury et al, based upon their experiences of designing assistive technologies for older people, propose extending the technical models of dependability to encompass the human element so that dependability can be rearticulated in terms of human aspects as well as the nature of error and faults. To this end they bring into modeling process matters such as fitness for purpose, adaptability, acceptability, and trustworthiness.

One way in which 'the human element' can be taken into account with regard to issues of dependability, or for that matter, other questions related to systems' design, that is stressed throughout this book is through the actual study of the work of people who will use the system and the study of how they use current systems. However, engaging in detailed ethnographic studies of work is time consuming and is also dependent upon the availability of good ethnographers, whether they are the designers themselves or dedicated professionals. In Chapter 7 Martin, Rouncefield and Sommerville address this issue by proposing a resource through which designers can systematically draw of the existing body of ethnomethdologically informed ethnographic studies of work (Randall et al 1995). One of the problems that designers face is the ability to find or draw generalities from out of particular studies, and the resource that Martin and Sommerville build is, in part, an attempt to provide a resolution to this problem.

To this end they detect patterns of what they call cooperative interaction. These are regularities, revealed by the corpus of studies, in the way in which work activities and interaction are organized. Martin and Sommerville identify a number of regular themes or topics that these studies have encountered: sequentially and temporality; a working division of labour; plans and procedures; routines, rhythms, patterns; coordination; awareness of work and ecology and affordances. The idea of 'patterns of cooperation' is that it is possible to generate generalized descriptions of interaction based upon specific studies of the various topics. Martin and Sommerville have so far documented ten such patterns. They have created a series of web pages that describe these patterns in a structured way, in each case moving from a high level description of the phenomena, 'The Essence of the Pattern', to three sections entitled 'Why?', 'Where Used', and 'Dependability Implications'. The reader can then drop into a vignette giving greater detail of the pattern and described in terms of the five topical headings. The idea is that designers can gain a quick insight into the social and interactional matters that might surround dependability issues for a system being designed for a situation that may correspond to one of the patterns of interaction. Simply, the patterns are thus a resource for considering dependability issues for new situations but which have similarities to the situation described in the patterns.

Martin and Sommerville are tackling a very complex, and for some, vexing issue in this chapter which is how to make studies of the workplace tell for situations not covered by a particular study and of making these studies a general resource for design as opposed to being a resource for a particular design for the situation studied. This is the first serious attempt to grapple with this issue and from the point of view of systems designers it is a welcome and important development.

While Martin and Sommerville are developing a radically new resource for the design of systems with respect to dependability, Dobson in chapter? considers the implications of a DIRC perspective on dependability for a more traditional design tool: modeling. Modeling is done to reduce the complexity of socio-technical systems, and Dobson describes how complexity can be handled by constructing different models of different parts of the system thus producing a suite of models. The distinctiveness of the approach is described by Dobson as residing in the fact that the models making up the suite related to one another within a conceptual framework, that of responsibility. Dobson's chapter is built up as a tutorial in such a modeling procedure. This chapter stands out from the rest of the collection because it is articulating a more usual tool in the methodological repertoire of design; nevertheless Dobson is using this tool to provide a DIRC type insight into dependability, for the conceptual framework for his models is the social matter of responsibility within an organization.

5. ETHNOGRAPHY AS A DESIGN METHODOLOGY

The strand of the DIRC project that most of the chapters in this book are drawn from is one within which the social considerations of systems' use predominate, and thus is the one within which the human science, especially sociological considerations are articulated. However, the Human Sciences in general, and sociology in particular, seethe with perspectival rivalry and methodological debate, something that can be clearly seen in Clarke et al's Chapter 1, in their review of different perspectives on the idea of trust. The fact that the Human Sciences is a battleground of competing perspectives means that the very idea within the DIRC program that a consideration of dependability issues in systems design should be grounded in the social world in which the systems will be used, is not a simple matter of turning to the appropriated discipline and using its findings. Computer scientists who have turned to sociology for insights into their problems are often surprised by the range of theoretical and methodological positions within sociology, and the intense disputes and rivalries between them. Given this situation then it might well be the case that turning to sociology to broaden an understanding of dependability might actually confuse matters, for there is no one social perspective that they could appeal to, there are a variety of them. For instance, from the point of view of labour process theory, making systems more and more dependable may be viewed as part of the general deskilling of labour under modern capitalism. From the point of view of postmodernism, however, dependability may be part of the objectification of society rendered by technological and scientific disciplines. While from yet another perspective, social constructionism, dependability may the product of rhetorical processes.

Sociology has often turned its attention to particular matters, for example, education, health, race relations, and the list of 'the sociology ofs…' is impressive. These subject areas, however, then become battle grounds on which the historical perspectival disputes of sociology are fought, and edited collections of sociological articles proliferate and articulate the various ways in which different sociological perspectives apprehend the phenomena. However, if we consider the point of view of systems designers rather than the point of view of the human scientist, this sort of internecine perspectival warfare may not be productive. Designers are not interested in a sociology of technology, and with understanding how different sociological factions reinterpret what they, the designers, do, from inside any particular sociological theory. The designers of systems who are interested in what sociology may offer have a very practical orientation. Thus, with respect to the issue of dependability they are interested in the way in which sociology can support them in tackling the issue of dependability as an engineering or design issue.

In this respect this current book, which is predominantly a sociological book, differs from the general run of sociological considerations of a phenomena. It is not attempting to provide different, sociological perspectives on, or push one sociological perspective about dependability in the manner of sociological collections on other topics might do. This is because it not *about* a sociology of design and engineering with regard to the issue of dependability, rather it is a sociology *for* design and engineering. The question this book raises is not about making dependability and trust topics for sociology, but how the way in which dependability and trust is articulated in the commonsense world of social relationships can inform design and engineering. This book is thus not about theories that provide different and competing interpretations of what, really, engineers are doing in building dependable systems. It is, rather, intended to have an actual impact upon the way in which designers consider building in the issue of dependability in the design of their systems. In this respect this book continues a research direction that has been established between some sociologists and the relevant design and engineering disciplines and which is to trade the analysis of work, and analysis of the use of technology, into the actual design of technology.

Those who have pursued this interdisciplinary research in general and those in this present collection in particular, have developed the relevancy of *ethnographically* gathered materials for design. This idea has been particularly promulgated in the field of Computer Supported Cooperative Work (CSCW) where ethnographers and computer scientists have been working together for more than a decade to articulate studies of work and organizations into computer systems' design. Ethnography, as practised within CSCW emphasizes the observation of work and technology use as it unfolds as a real time phenomena, and the apprehension of the participants' point of view. As a way of gathering data it stands in contrast to surveys and questionnaires. Given designers practical interests, it is not surprising that the field work methodology of ethnography has interested them more than the theoretical or statistical strands of the human sciences. Ethnography emphasizes investigating matters of work and use empirically, as opposed to theoretically; in real worldly circumstances, as opposed to contrived experimental situations; in real time, as opposed to generalized time; and as work and use unfolds, as opposed to after the fact stories about work. This gives designers a further methodology through which to develop requirements for systems and to assess systems in use. Many of the authors of this book have been at the forefront of developing ethnography as a methodology for design in the field of CSCW, and through this book all they are showing its relevancy for yet another area of systems design, that of dependability.

However, ethnography is not all of a piece. There are a number of sociological positions that gather materials through the fieldwork of ethnography: symbolic interactionism, social studies of science and technology, and ethnomethodology being some. What these different sociological positions then do with those materials can, however, be very different. For example, within social studies of science and technology there is an emphasis on understanding how the science or the technology is a construction of social processes, while in ethnomethodology there is an emphasis upon the uniquely adequate features of work. It is a curious fact, however, that the ethnographers within CSCW tend to emphasize ethnomethodology, and this is also reflected in this book, for the position adopted by most, though importantly not all, within this collection is also that of ethnomethodology. This is not something that, in the main, is overtly announced, but it is discernable in the character of most of the studies, and in the invocation of other relevant studies. In one respect it may not matter to the designer that this is ethnomethodology; it is just the utility of the sociology for design purposes that is important. In another respect it is, for the utility may actually rest upon the character of the study as ethnomethodological. An introduction to a collection is not the place, however, to explore this matter, and readers can now turn to the actual chapters themselves to start to form their own opinions.

REFERENCES

1. Bowers, J., Button, G. and Sharrock, W.W. (1995). Workflow From Within and Without: Technology and Cooperative Work on the Print Industry Shopfloor, in H. Marmolin, Y. Sundblad, and K. Schmidt. *Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work* Dordrecht: Kluwer Academic Publishers.
2. Garfinkel, H. (1963). "A conception of, and Experiments with, 'Trust' as a Condition of Stable Concerted Actions. " In *Motivation and Social Interaction.* Edited by O.J. Harvey, 187-238. New York: The Ronald Press.
3. Garfinkel, H. (1967) *Studies in Ethnomethodology* . Englewood Cliffs, NJ: Prentice- Hall.
4. Heath, C. and Button, G. (eds) (2002). Special Issue on Workplace Studies, *The British Journal of Sociology*, Volume 53, Number 2.
5. Randall, D., Rouncefield, M. and Hughes, John (1995) Chalk and Cheese: BPR and Ethnomethodologically Informed Ethnography in CSCW. In H. Marmolin, Y. Sundblad and K. Schmidt (eds.) *Proceedings of the Fourth European Conference On Computer Supported Cooperative Work.* Dordrecht: Kluwer Academic Publishers.
6. Livingston, Eric (1986). *The Ethnomethodological Foundations of Mathematics*. London: Routledge & Kegan Paul.
7. Luff, P., Hindmarsh, J. and Heath, C (eds) (2000). Workplace *Studies: Recovering Work Practice and Informing Design*. Cambridge: Cambridge University Press.
8. Suchman, Lucy (1994). Do categories have politics?: The Language/Action Perspective Reconsidered, *Computer Supported Cooperative Work* 2.
9. Sudnow, D. (1993). *Normal Crimes: Sociological Features of the Penal Code in a Public Defender Office*. Ivington Publishers.

Chapter 1

TRUST AND ORGANISATIONAL WORK

Karen Clarke[1], Gillian Hardstone[2], Mark Hartswood[2], Rob Procter[2] and
Mark Rouncefield[1]

1 Department of Computing, Lancaster University, UK.
2 Institute for Communicating and Collaborative Systems, Division of Informatics, University of Edinburgh, UK.

"For most of us, most of the time, our natural attitude in the taken-for-granted world is one which enables us to maintain our sanity in our passage through life and the daily round. Routines, habits ...and the consistencies with which our interactions with each other conform to expectations, together provide the infrastructure for a moral universe in which we, its citizens, can go about our daily business. Through learning to trust others we learn, one way or another, to trust things. And likewise, through learning to trust material things we learn to trust abstract things. Trust is therefore achieved and sustained through the ordinariness of everyday life and the consistencies of both language and experience." (Silverstone)

".. there is no relationship of trust with a computer "(Shneiderman 2000)

1. INTRODUCTION: NOTIONS OF TRUST

"Without trust only very simple forms of human cooperation which can be transacted on the spot are possible ... Trust is indispensable in order to increase a social system's potential for action beyond these elementary forms"(Luhmann 1990)

K. Clarke, G. Hardstone, M. Rouncefield and I. Sommerville (eds.), Trust in Technology: A Socio-Technical Perspective, 1–20.