Ethnographically-informed systems design for air traffic control

R. Bentley[*], J. A. Hughes[†], D. Randall[†], T. Rodden[*], P. Sawyer[*], D. Shapiro[†], I. Sommerville[*]

CSCW Research Centre
Computing[*] and Sociology[†] Departments,
Lancaster University,
Lancaster, LA1 4YR
UK.
E-mail: is@comp.lancs.ac.uk
Phone: +44-524-65201
Fax: +44 524 381707

ABSTRACT

This paper relates experiences of a project where an ethnographic study of air traffic controllers is being used to inform the design of the controllers' interface to the flight data base. We outline the current UK air traffic control system, discuss the ethnographic work we have undertaken studying air traffic control as a cooperative activity, describe some of the difficulties in collaboration between software developers and sociologists and show how the ethnographic studies have influenced the systems design process. Our conclusions are that ethnographic studies are helpful in informing the systems design process and may produce insights which contradict conventional thinking in systems design.

KEYWORDS

Air Traffic Control, Systems Design, Ethnography

INTRODUCTION

One of the most significant influences of CSCW has been to lend a respectability amongst software developers to sociological methods of empirical research. The contribution of sociology, and anthropology, is through the description and analysis of the 'real world' settings in which systems will have to find their place. The social world of work is so rich and varied as to defy adequate representation by individualised, cognitive models of users. The notion of the user as a stable generic entity with fixed cognitive capacities is a chimera, and a model which cannot survive the move toward larger scale, distributed systems. Also important is the growing realisation that one of the major causes of system 'failure' is the mismatch between the functionalities of the system as conceived by designers and its contexts of use [5].

Seminal work in this area has been carried out by Suchman [12] and by other investigators such as Hughes et al. [9], Harper et al. [6] and Heath and Luff [7]. These ethnographic studies have involved a sociologist observing workers in their environment over a period of several months and hence gaining a deep understanding of the actual rather than the formal working practices. For example, Suchman has studied office systems, Heath and Luff have studied a control room for the London Underground railway and Hughes et al. studied air traffic control.

The work reported in this paper is a continuation of this last study. It is distinct from previous ethnographic studies of controllers in that the explicit objective of the ethnography is to inform the design of a user interface to a reactive data base system which provides the essential information for air traffic controllers to carry out their work. A reactive data base is a data base where the information is continually updated either from external sensors (radars and aircraft data links in this case) or from concurrent inputs from different users.

There have been several previous experiments with automating the user interface to the flight data base system so that instructions issued by a controller are captured in the data base. Apart from speeding up the communication between controllers, computer-processable information about controllers' intentions could be used to support an effective conflict alert facility. However, given the incremental style of innovation which dominates in air traffic control, any technical change must require minimal changes to working practices not only to avoid the huge cost of retraining but also because the data base interface is only a part of the complex of systems that is air traffic control.

Previous experiments in automating the UK system have been undertaken primarily from a technical viewpoint. They have not been acceptable to air traffic controllers and we believe that this is because these systems have not effectively supported the 'working division of labour'

[1] as adopted by controllers. Hopkin [8] identifies the problem:

> "One striking aspect of automation applied to air traffic control systems is that most of the forms of automation for the controller to use, as distinct from those which sense or process or compile data automatically, are for one controller at a human-machine interface. They are aids to an individual controller's decisions, problem solving or predictions, yet they are being introduced into contexts where many of these functions have previously been performed by teams"

Our work is different. We are starting from the position that air traffic control is a subtle cooperative activity and we believe we must understand the nature of that cooperation in order to build effective computer support.

This paper is not concerned with the details of the automated system which is under development and which is described elsewhere [2, 3]. Rather, we discuss how the ethnographic studies have helped us gain an understanding of the cooperative processes of air traffic control and how this understanding has influenced the design of our prototype software system. In the remainder of this paper we briefly describe the current system for UK air traffic control and the cooperation in the air traffic control process. We describe some problems of cross-disciplinary working and discuss how the ethnographic studies have led us to question some widely-held assumptions about systems design.

THE UK AIR TRAFFIC CONTROL SYSTEM

Air traffic control in the UK is managed through two sites which control all domestic and international air traffic. Our studies were concerned with the larger of these sites, the London Air Traffic Control Centre (LATCC). Within LATCC, there are 8 radar suites each of which is responsible for controlling 1 or more sectors of the airspace. Each suite provides a working area for two teams of controllers (one team per sector), radar screens and facilities for communications between aircraft and other suites. The radar is a 2-dimensional, 'real time' representation of the sector's airspace and the positions of aircraft within it. It can display other information about a controlled sector including sector boundaries, coast lines, major and minor airports, etc. However, the radar only shows what is 'happening now' but not what 'might be happening' in a few minutes time. Controllers need to anticipate potential problems so, to supplement the radar, they use paper *flight progress strips* which carry information about expected and current flights being controlled, together with controllers' instructions to the controlled aircraft. The ATC system is described in more detail by Hughes *et al.* [10] who discuss the ethnographic studies of controllers and who describe flight strip layout more fully.

An example of a flight strip (as replicated in our automated system prototype) is illustrated in Figure 1. The strip contains static information such as the flight number, aircraft identifier, beacon identifier, source and destination, and dynamic information such as time over beacon, heading, current height and airspeed.

Strips are produced from a data base consisting of flight plans filed prior to the aircraft's departure. The data base is updated through Radar Data Processing (RDP) and by controllers and their assistants directly inputting flight data.

Flight strips are organised on a *flight progress board* where strips are aligned and organised in a rack according to the reporting points over which a flight will pass (Figure 2). To the accomplished controller this display is an 'at a glance' means of showing the flow of traffic through the sector and its characteristics. To the experienced and knowledgeable eye, different indicated ETA times and different heights may represent problems, for example, 'climbing through' or 'catching up' given the relative performances of aircraft. The way in which the controller organises the strips, for instance, according to arrival time over a reporting point, or according to flight level, or to possible confliction points, provides information about the state of the sector.

The controller issues instructions to aircraft and verifies, using the radar, that the instructions have been carried out. Controllers coordinate the air traffic safely by maintaining separation levels, directing course changes, airspeed, ascents and descents. They have to take the traffic as it comes and blend incoming aircraft into a steady flow. To this end, they spend time organising strips into a sequence of aircraft, using the radar to check and assess the current 'state of play' and, using the strips, anticipate likely problems. They also spend time coordinating activities with other members of the team, arranging flight levels, discussing 'best' procedures to expedite the flow of traffic, sometimes passing flight level changes to assistants to coordinate with adjacent sectors, and so on.

The active organising of the strips themselves on the flight progress board by the controller is an essential feature of the work. At its simplest and most general, the controller's problem is a scheduling one. For any segment of airspace the traffic has to be taken as and when it arrives and threaded into an orderly pattern before each individual plane is handed on to the next sector. This scheduling has to be achieved in and through making the traffic flow. Aircraft cannot be parked and even when they are 'on hold' they are still on the move and part of the traffic flow. Organising the strips is a way of achieving a solution to the scheduling problem by organising the strips as a scheduling of work tasks.

| 9.37 | BTN | 280 ↑ | BRITANNIA
BAL770
5423
M/B737/C   T420 | 300

EGGW UA2 UB3 UB4 EGAA | CREWE   9.26 |

Figure 1 A flight strip

Barton Flight Strip Bay

9.37	BTN	280 ↑	BRITANNIA BAL770 5423 M/B737/C T420	300 EGGW UA2 UB3 UB4 EGAA	CREWE 9.26
9.15	BTN	■ 370	KLM KLM604 1144 M/B747/C T469	155 KLAX UB3 UB4 UB105 FHAM	IOM 9.08
9.07	BTN	330 ↓	DANAIR DAN081 5123 M/BA11/C T410	300 EGPD UA1 EGKK	CREWE 8.57
8.58	BTN	■ 370	JETSET AMM944 1172 M/B757/C T442	175 EGPE UA25 UA28 UB107 ***	DCSX 8.48

Figure 2 Flight strips representing aircraft due to arrive at the Barton beacon organised by arrival time

COOPERATION IN THE ATC PROCESS

The air traffic control system is deliberately organised to minimise explicit coordination and cooperation between controllers. For example, so long as flights are as planned, the handover of a plane from a controller in one sector to a controller in an adjacent sector does not require any communication between them. A task-based systems analysis would therefore fail to reveal the subtle and complex cooperation which is actually going on. This cooperation only became apparent through the ethnographic studies.

A typical radar suite is manned by a chief controller, two radar controllers, and two assistants with military liaison officers in close attendance. Each of these is responsible for particular tasks, though there is a very important informal sharing of some of them. Assistants perform apparently routine servicing tasks, such as tearing off the printed strips from the printer, placing them into their correct coloured holders, ensuring that each strip is placed in the correct rack. However, during this action, they informally check the information on the strip. Errors are not uncommon and this initial checking is an important first level check. For example, in our observations, we noticed an assistant controller recognise that a flight destination was obviously wrong. As he glanced at the strip, he saw that the destination airport specified was a private manufacturer's airfield yet the flight was a normal scheduled flight. From his knowledge of that specific flight, he corrected the strip before passing it to the controller. Assistants may also update flight plans by editing route and ETA changes.

The radar controllers are responsible for the actual control instructions issued to the aircraft and the chief's responsibilities normally include coordinating traffic between sectors by telephone calls or by visits to the relevant sector suite. His or her main task, however, is monitoring the flow of traffic at a strategic level, making judgements about the allocation of flight levels, and so on, to facilitate the traffic flow. Chiefs may also act as a backup to the radar controller by dealing with unscheduled traffic, drawing attention to potential problems by 'cocking out' strips (setting them at an angle in the rack), or by writing on them. The chief also coordinates military aircraft with the military liaison officers.

What is important about the teamwork is not so much that the respective team members have individual tasks to perform, which they do, but that they are woven into a 'working division of labour'. This means the division of labour is organised dynamically according to need and does
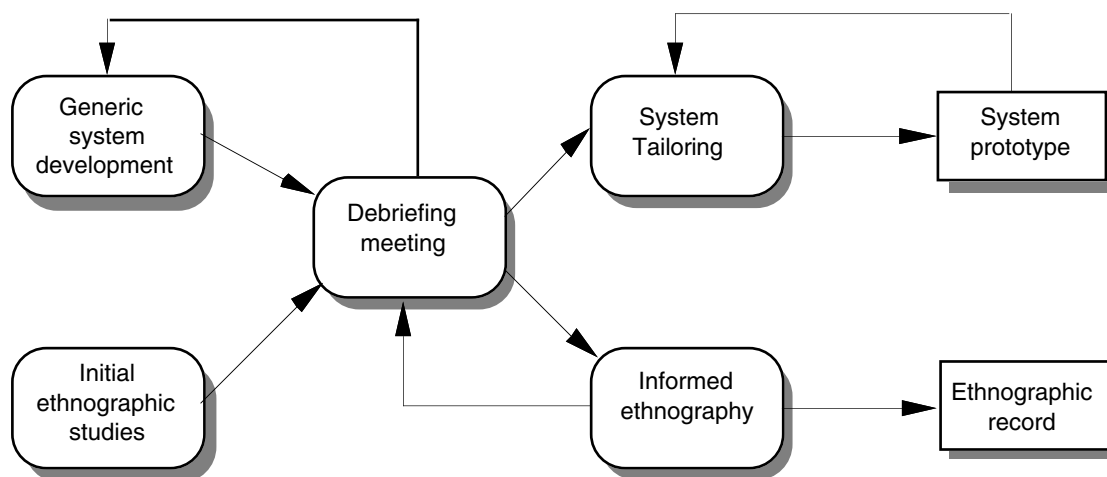
Figure 3 Ethnographically-informed systems development

not necessarily follow a prescribed form. The individuals are individuals-in-a-team, and much of their work consists in the ability to organise the distribution of individual tasks into an ongoing assemblage of activities *within* the 'working division of labour'.

Our ethnographic studies revealed that the teamwork character of the work was manifested in myriad ways in the ongoing work itself. Not as an adjunct to the work but integral to it. There was no one thing, or set of things, that manifested teamwork since it saturated the work itself - in the talk around the suite, in the activities performed, in the stories and anecdotes told, and more. Indeed, this was much the way in which the ethnography recorded the work, by stories, descriptions of activities, anecdotes, vignettes of events, and so on. But teamwork is very much an aspect of its members' tacit knowledge which is not normally explicitly expressed except on the occasions when 'normal and routine' working breaks down.

Effective use of flight strips is the key to 'good controlling'. As one controller put it, 'You have got to have a complete picture of what should be in your sector and what should be in your sector is on those strips'. Strips become a documentary record of an aircraft's passage through the sector. The initial information on the strip is amended as instructions are given to and confirmed by the pilot. The strip accumulates information about actions which have been recorded on the strip. Essentially, the strip is a shared note pad conveying to members of the team what actions have been taken with respect to particular aircraft, who authorised these actions and how these might affect other aircraft in the traffic configuration.

Amendments may be done by the controller, by the chief or, less often, by an assistant. Attention-getting information may also be written on the strips, such as arrows indicating unusual routes, symbols designating 'crossers, joiners and leavers', circles around unusual destinations, and so on. Information indicating that coordination between sectors has taken place, changes in ETA or to route, changes in call sign, etc., can be indicated using standard notations, each member of the team using a different coloured pen to annotate the strips,

thus adding responsibility and accountability to the record. The strip, in other words, is a public document for the members of the team; a working representation of an aircraft's control history and a work site of controlling.

WORKING TOGETHER: ETHNOGRAPHERS AND SOFTWARE ENGINEERS

The challenge which we faced was how to utilise the insights of the ethnographic study of air traffic controllers in the design of user interfaces to a flight data base which would provide comparable functionality to that provided by the flight strip system. Furthermore, because of the time constraints on our project, we could not carry out an ethnographic study then develop the software system; rather the ethnography and the systems development had to be concurrent activities. The approach we have adopted is illustrated in Figure 3.

The debriefing meetings are a particularly important part of the process. During these meetings, the ethnographer discussed his findings and was questioned by other team members. The software developers' questions focussed on systems requirements and, while it was rare to identify an explicit software requirement during the debriefing meeting, the developers gained an intuitive impression of the facilities required by controllers.

During the debriefing meetings, the system developers identified particular areas of interest and particular problems which should be investigated in the next phase of the ethnography. Thus, the ethnographer was informed of the system requirements and focussed his observations to answer the questions posed by the system developers.

In the first phase of our work, we have developed a prototyping system [2] which has been designed so that we can rapidly generate user interfaces to a reactive data base. In the second phase, which is now underway, we are experimenting with different approaches to user interface design. The ethnographer plays a key role in this activity as he acts as a substitute for the air traffic controller, and represents his or her view of the system. This allows us to have some of the advantages of participative design but without the requirements that an end-user should be regularly available for system evaluation. In the final

phase, we will expose the system to real controllers and evaluate the interfaces which we have designed.

In our work, we have had the advantage that the people involved from both sociology and software engineering were willing to be flexible and willing to recognise that the other discipline's viewpoint may be valid irrespective of how alien it might be. In this respect, we believe our collaboration has been successful but there have been difficulties which we believe are likely to arise in other collaborative projects of this nature.

1. The fundamental approach of each discipline is totally different. Sociology is analytic. It is concerned with gathering and interpreting data about some social situation or process and drawing some conclusions from that interpretation. By contrast, software engineering is concerned with synthesis - designing and building new abstract models of the real-world. Thus, sociology focuses on and pays great attention to detail; software engineering strives to hide detail through abstractions.

2. Social analysis is a fairly prolonged activity. Typically, ethnography will take place over a period of several months with at least the same amount of time spent in analysis and interpretation of the observations. To inform the systems development process, however, software engineers need quick results and make demands on the sociologists to provide rapid assessments of their work.

3. The roots of ethnography are in anthropology so ethnographers are trained to avoid making judgements about a social situation or process and, as far as possible, to avoid letting their own prejudices interfere with their observations. By contrast, engineers *must* make judgements, often with inadequate information, as to what is and is not significant. Engineers therefore may find the more remote attitude of sociologists difficult to understand.

These differences in working styles were highlighted by four questions which systems designers posed to the ethnographer studying air traffic control. These questions were motivated by the need to make design decisions about the system which was being built to support prototype generation and the data base interfaces themselves:

- what characteristics of the existing manual system are unimportant and need not be supported in an automated system?
- what are important manual activities which need not be supported in an automated system because the activities are a consequence of the fact that no automated support is available?
- what characteristics of the manual system must be replicated without change in an automated system?
- what activities from the manual system may be supported in a way which is different from that used in the manual system?

From the point of view of the sociologists, what was interesting about such questions was their directedness in the sense that almost for the first time the service role of the sociological analysis was explicitly formulated. The questions were not easily answerable, not least with reference to a way of working as subtle and as complex as controlling. Moreover, the sociologists' own methodological point of view which treated the system as a fusion of working practices and technology made it difficult to draw the distinctions necessary to answer such questions. For example, deciding what were the 'important manual activities', irrespective of the availability of automated support, was not straightforward. Was 'idle chat' among the team 'unimportant', and in what sense? Even though such talk might not be directly related to the specific tasks of controlling, a case could be made that it was important for morale, the sharing of experiences, providing support, and so on. Similarly, although a number of manual activities of the 'manual system' may be supported in different ways, it is difficult to anticipate the likely consequences of such a shift.

Moreover, the actual performance of some manual activities, such as writing on the strips, manipulating them in the racks, coordinating by telephone with adjacent suites, and more, serve to keep the controller, and other members of the team, 'geared into' the work. Though some of these could be automated, the important question of whether this would reduce the controller to a more passive role of system monitor rather than an active participant in a system-in-use is hard to determine in advance.

The sociologists have recognised the reason why these questions need to be answered; the software engineers have recognised the difficulties in providing such answers. We believe that finding a means to answer questions such as these is the key to the systematic use of ethnography in the systems design process.

ETHNOGRAPHY AND SYSTEMS DESIGN
Our work has now reached a stage where we are generating system interfaces whose design has been informed by the ethnographic observations. We have found that the information provided by ethnography is essentially background information which has provided a deeper understanding of the application domain. The ethnography did not result in specific, detailed systems requirements; rather it provided pointers to appropriate design decisions.

From our observations we have become convinced that some 'conventional' assumptions made by systems designers may be invalid when cooperative systems are being developed. Examples of these assumptions are:

1. Computer systems should always automate tedious manual tasks which involve comparisons of similar information and ordering of records in a data store. Therefore, the computer system not the human operator should be responsible for sorting and maintaining the sort order when new information is added to the system.

2. User interface designers should always provide facilities for end-users to tailor interfaces to suit their own personal ways of working and personal preferences.

Our ethnographic observations revealed that the manual manipulation of flight strips and the manual re-ordering of the flight strip racks were significant activities. To provide an analogue of this manual manipulation, our prototype system incorporates the following design decisions:

1. Strips are not positioned according to some default order when they are added to a display.

2. Some activities have been deliberately retained as requiring manual intervention because the manual activity allows a feasibility check of the strip information to be made.

3. There is no standard, enforced model of flight progress board organisation. Support is provided for many different possible orderings or ordering as specified by the controller.

When a controller is using strips ordered by time of arrival to a beacon and new strips have to be added to the display, it would seem natural for these strips to be positioned automatically in the right place on the user's display. However, the ethnographic work suggested that the controller's action of manually placing a strip in the appropriate position in a bay focussed attention on that strip and to related strips. This was an important safety device as it allowed the early identification of potential problems. Automated positioning is undesirable because these problems may not become apparent until there is an explicit need to use the new strip.

A further example of manual intervention forcing data checking occurs when strips are first printed in the current system. They are then placed into different coloured holders by assistant controllers to distinguish between flights on different headings. In this action, the assistants read the strip information and detect irregularities such as known flight numbers flying to the wrong airport, an arrival time which is inconsistent with the aircraft type, etc. They either correct these errors themselves or draw the controller's attention to them.

The colour of a strip holder is determined by the aircraft's heading and it would be a simple matter to automate the assignment of coloured borders to electronic strips. However, because of the important checking function, we do not propose any default border colour but require the controller to make a manual choice of possible display formats for a strip.

The manual of air traffic control specifies that strip bays should normally be ordered according to the time of arrival of an aircraft over a beacon. This model is followed by the majority of controllers but our ethnographic observations revealed that some controllers have evolved an alternative way of working which uses a different strip ordering. Therefore, while we provide explicit facilities which allow a controller to order his or her display according to the manual, we also immediately switch off this automatic ordering as soon as strips are manually re-ordered.

It is a common belief amongst developers of user interfaces that end-user tailorability is essential. Indeed, Fischer and Girgensohn [4] state:

"End-user modifiability is not a luxury, but a necessity in cases where systems do not fit a particular task, a particular style of working or a personal sense of aesthetics".

While this may be true for applications designed for single-user use, we do not think it valid for cooperative systems where the representation is shared with a common understanding of the syntax and semantics of that representation. Our work has shown the importance of such a common representation for communication amongst controllers. Much of their work requires 'at a glance' observations of strips and flight progress boards. This can only be effective if all controllers can rapidly assimilate flight strip information and this rapid assimilation is hindered if even slight differences in strip representations are supported. We have therefore limited tailorability to surface features such as highlighting and have curtailed the controller's ability to tailor the strips in such a way that extensive explanation to other controllers would be required.

As well as influencing the systems design process, we believe that the ethnographer has a further role as substitute user during initial system validation. User-centred design where users participate in the interface design process from an early stage in that process is likely to lead to more effectively usable user interfaces. However, a serious constraint to the practice of user-centred design is the availability of users, particularly where these users are an expensive and scarce resource. This is a particular problem during early stages of the design process where the design is unlikely to satisfy the user's requirements so the user gets little reward from participation in the design process.

Because of the understanding of the user's work which is gained by the ethnographer in the course of his or her studies, our experience has shown that ethnographers can act as 'user's champions' in the early stages of the design process. Thus, initial inappropriate designs can be detected with very limited end-user involvement so that expensive user time is only used at later stages of the process where design details have to be resolved. This issue is discussed in more detail by Sommerville *et al.* [11]

CONCLUSIONS

Our general conclusions about our collaboration and our studies of air traffic controllers are as follows:

1. Observational methods such as ethnography can play an important role in informing the process of automated systems design. However, there is not an obvious mapping from an observational record and a systems requirements document. The extent of the contribution which observational methods can make to the system specification process has still to be demonstrated.

2. Software engineers and sociologists can work together effectively. However, there is a wide gulf between these disciplines and entrenched philosophical positions will probably ensure that that gulf cannot be bridged. Effective inter-disciplinary cooperation requires much flexibility on both sides and requires both sides to question their own assumptions and working methods.

3. Some conventional principles which are normally thought of as 'good design' may be inappropriate for cooperative systems. Manual intervention and manipulation of information may be essential implicit methods of communication and cooperation.

4. An important role for ethnographers is to act as substitute users in a 'user-centred' systems design process. Because of their close involvement with end-users, ethnographers are well equipped to understand their problems and can be effective in discovering gross errors in the systems design.

The collaboration on studies of air traffic control is continuing with evaluation studies planned for 1993. A further project is also underway where a much less structured cooperative activity (systems design) is being studied and we anticipate that it will be possible to draw interesting comparisons between these projects.

REFERENCES

[1]   Anderson, R. J., Hughes, J. A. and Sharrock, W. W. 1989, Working for Profit: The Social Organisation of Calculability in an Entrepreneurial Firm, Aldershot: Avebury.

[2]   Bentley, R., Rodden, T., Sawyer, P. and Sommerville, I., 1992, 'A Prototyping Environment for Dynamic Data Visualisation', To appear in the Proceedings of the 5th IFIP Working Conference on User Interfaces, (Ellivuori, Finland), Elsevier Science, 10-14 August, 1992.

[3]   Bentley, R., Rodden, T., Sawyer, P. and Sommerville, I., 1992, 'An architecture for tailoring cooperative multi-user displays', To appear in the Proceedings of CSCW'92.

[4]   Fischer, G. and Girgensohn, A., 1990, 'End-User Modifiability in Design Environments', Proceedings of CHI '90, (Seattle, Washington), ACM Press, 183-191.

[5]   Grudin, J., 1991, 'CSCW: The Convergence of Two Development Contexts', Proceedings CHI'91, 91-98, Reading, Mass.: Addison Wesley.

[6]   Harper, R., Hughes, J. and Shapiro, D., 1991, 'Harmonious Working and CSCW: Computer Technology and Air Traffic Control', in J. M. Bowers and S. D. Benford (eds): Studies in Computer Supported Cooperative Work. Theory, Practice and Design, North-Holland, Amsterdam.

[7]   Heath, C. and Luff, P., 1991, 'Collaborative Activity and Technological Design: Task coordination in the London Underground Control Rooms', in Proceedings of ECSCW'91, Kluwer.

[8]   Hopkin, V. D., 1991, 'The Impact of automation on Air Traffic Control Systems', in. J. A. Wise, V. D. Hopkin and M. L. Smith (eds.), Automation and Systems Issues in Air Traffic Control, Berlin: Springer-Verlag.

[9]   Hughes, J. A., Shapiro, D. Z., Sharrock, W. W., Anderson, R. J., Harper, R. R. and Gibbons, S. C., 1988, 'The Automation of Air Traffic Control', Final Report SERC/ESRC Grant no. GR/D/86257, Department of Sociology, Lancaster University.

[10]   Hughes, J. A., Randall, D. W. and Shapiro, D. Z., 1992, 'Faltering from Ethnography to Design', To appear in Proceedings of CSCW'92.

[11]   Sommerville, I., Rodden, T., Sawyer, P. and Bentley, R., 1992, 'Sociologists can be Surprisingly Useful in Interactive Systems Design', To appear in Proceedings of HCI'92, York, UK, September 1992.

[12]   Suchman, L., 1987, Plans and Situated Actions, Cambridge: Cambridge University Press.