Creative knowledge work and interaction design

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfillment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/6992

Publisher: © L. Candy

Please cite the published version.
This item is held in Loughborough University’s Institutional Repository (https://dspace.lboro.ac.uk/) and was harvested from the British Library’s EThOS service (http://www.ethos.bl.uk/). It is made available under the following Creative Commons Licence conditions.

![Creative Commons Licence](https://creativecommons.org/licenses/by-nc-nd/2.5/)

**Attribution-NonCommercial-NoDerivs 2.5**

You are free:

- to copy, distribute, display, and perform the work

Under the following conditions:

**Attribution.** You must attribute the work in the manner specified by the author or licensor.

**Noncommercial.** You may not use this work for commercial purposes.

**No Derivative Works.** You may not alter, transform, or build upon this work.

- For any reuse or distribution, you must make clear to others the license terms of this work.
- Any of the above conditions can be waived if you get permission from the copyright holder.

Your fair use and other rights are in no way affected by the above.

This is a human-readable summary of the [Legal Code (the full licence)](http://creativecommons.org/licenses/by-nc-nd/2.5/).

For the full text of this licence, please go to:
http://creativecommons.org/licenses/by-nc-nd/2.5/
CREATIVE KNOWLEDGE WORK AND Interaction Design

by Linda Candy

A Doctoral Thesis

Submitted in partial fulfilment of the requirements
for the award of

the Degree of Doctor of Philosophy of Loughborough University

March 1998

© L. Candy 1998
Acknowledgements

There are many people who, in their different ways, have enabled me to carry out the research described in the publications that are presented in this thesis. I am very grateful for the generous time given for interviews and discussions by the subjects of the case studies: in particular, Stella O’Brien, Mike Burrows and Ken Sears of Lotus Engineering should be mentioned. For the indispensible technical support that research demands, I am immensely thankful to the support staff of the Computer Science department at Loughborough University, and, in particular, to André Schappo. I am also indebted to the members of the LUTCHI research teams, both past and present, who gave me the benefit of critical appraisal and argument. My Director of Research, Chris Hinde has guided me in the drawing together of the thesis and has given much needed feedback. Ernest Edmonds, who first encouraged me to pursue research in this field, has provided invaluable insights and that indispensible ingredient, constructive criticism. Finally, the support of my family and friends has been unstinting, for which I can never thank them enough.
Dedicated to Emma, Meroë and Robert
Abstract

The main aim of the research presented in this thesis is to inform the design of interactive computer systems for supporting creative knowledge work. Research into creativity and knowledge work has been explored and used to develop a criteria modelling approach. The particular contribution of the author’s work is the drawing together of that research and applying the findings to interaction design. The publications were selected on the basis of how well they represent the main outcomes of the work. The journey from prescribing system requirements and design goals to framing the system design process in terms of evaluation criteria may be traced through the papers presented.

Interest in creativity and the role of computer technology in creative tasks has recently increased. A number of national initiatives have been set in motion in the UK, beginning in December 1996 with the Initiative for National Action on Creative Technologies, the Creative Media Initiative: Technology Foresight, Department of Trade and Industry, National Endowment for Science and Technology in the Arts (NESTA) and the People and Computers Programme, of the Engineering & Physical Science Research Council (EPSRC). Thus, the author’s involvement in creativity research and computer support is proving to be timely. Amongst her recent initiatives is Creativity and Cognition, an international symposium which brings together creative people in the arts with technologists and scientists.

The thesis is divided into three parts: themes and outcomes, methodology and case studies. A criteria-based modelling approach is presented which has evolved from earlier models that represent key elements of creativity and knowledge work. A model of creative knowledge work is proposed and categories of criteria identified. Underpinning the main outcomes are the case studies which were carried out in industry/academic collaborative projects. The findings were considered in relation to other studies. The thesis presents an approach to computer systems design and development that directly links the requirements definition to the application of evaluation criteria. These criteria are based upon the characteristics of the cognitive style and working practices of creative knowledge workers.
<table>
<thead>
<tr>
<th>Chapter 1</th>
<th>Thesis Overview</th>
<th>6-28</th>
</tr>
</thead>
<tbody>
<tr>
<td>Part 1</td>
<td>Themes and Outcomes</td>
<td></td>
</tr>
<tr>
<td>Chapter 2</td>
<td>Creativity, Knowledge and Computer Support</td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>Computers and Creativity Support: Knowledge, Visualisation and Collaboration</td>
<td>29-57</td>
</tr>
<tr>
<td>2.2</td>
<td>Creativity in Knowledge Work: A Process Model and Requirements for Support</td>
<td>58-74</td>
</tr>
<tr>
<td>2.3</td>
<td>Interactive Knowledge Support to Conceptual Design</td>
<td>75-97</td>
</tr>
<tr>
<td>Chapter 3</td>
<td>Criteria-Based Modelling and Interaction Design</td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td>Supporting the Creative User: A Criteria-Based Approach to Interaction Design</td>
<td>98-110</td>
</tr>
<tr>
<td>3.2</td>
<td>Cognitive Modelling of Creative Knowledge Work for Interaction Design Criteria</td>
<td>111-138</td>
</tr>
<tr>
<td>Part 2</td>
<td>Methodology</td>
<td></td>
</tr>
<tr>
<td>Chapter 4</td>
<td>Computer System Life Cycle Models</td>
<td></td>
</tr>
<tr>
<td>4.1</td>
<td>Computers and the Curriculum: An Action Research Perspective</td>
<td>139-152</td>
</tr>
<tr>
<td>4.2</td>
<td>The Twin Paths of Research and Design: Reformulating the Process</td>
<td>153-170</td>
</tr>
<tr>
<td>Chapter 5</td>
<td>Computer System Development Strategies and Methods</td>
<td></td>
</tr>
<tr>
<td>5.1</td>
<td>Issues in the Design of Expert Systems for Business</td>
<td>171-197</td>
</tr>
<tr>
<td>5.2</td>
<td>A Strategy and Technology for FE System Development</td>
<td>198-210</td>
</tr>
<tr>
<td>Part 3</td>
<td>Case Studies</td>
<td></td>
</tr>
<tr>
<td>Chapter 6</td>
<td>Study 1: A Business Applications of Expert Systems</td>
<td></td>
</tr>
<tr>
<td>Chapter 7</td>
<td>Study 2: Expert Interaction with a Speech Knowledge System</td>
<td></td>
</tr>
<tr>
<td>7.1</td>
<td>End User Manipulation of a Knowledge-Based System: A Study of an Expert's Practice</td>
<td>224-247</td>
</tr>
<tr>
<td>Chapter 8</td>
<td>Study 3: Innovative Product Design and Creative Cognition</td>
<td></td>
</tr>
<tr>
<td>8.1</td>
<td>Artefacts and the Designer's Process: Implications for Computer Support to Design</td>
<td>248-269</td>
</tr>
<tr>
<td>8.2</td>
<td>A Study of the Creative Design of the Lotus Bicycle: Implications for Knowledge Support System Research</td>
<td>270-293</td>
</tr>
<tr>
<td>Chapter 9</td>
<td>Study 4: Engineering Design Management and Design Models</td>
<td></td>
</tr>
<tr>
<td>9.1</td>
<td>Representations of Strategic Knowledge in Design</td>
<td>294-318</td>
</tr>
<tr>
<td>Appendix 1</td>
<td>Full List of Publications</td>
<td>319-323</td>
</tr>
<tr>
<td>Appendix 2</td>
<td>Contribution of Joint Authors</td>
<td>324-325</td>
</tr>
<tr>
<td>Appendix 3</td>
<td>Criteria Sources List</td>
<td>326-332</td>
</tr>
</tbody>
</table>
CHAPTER 1 THESIS OVERVIEW

1. Introduction

The main aim of the research presented in this thesis is to inform the design of interactive computer systems for supporting creative knowledge work. Research into creativity and knowledge work has been explored and used to inform the direction of the modelling approach and its relevance to interactive computer system design.

The thesis represents the culmination of the research outcomes from a series of case studies conducted by the author. This work has arisen from funded research projects spanning diverse applications of technology but with a common theme, that of the design requirements of interactive computer systems that meet the needs of professional, expert users carrying out complex tasks.

Fourteen papers have been selected from a larger set of fifty publications (see Appendix 1 for full list). Most papers prior to 1997 are jointly authored, a fact which reflects the multidisciplinary nature of the research projects from which they arose in the field of Human-Computer Interaction (HCI). A note delineating the different contributions of the other authors is made in Appendix 2.

The particular contribution of the author’s work is the drawing together of empirical research in the area of creativity and knowledge work in a number of expert domains and applying the results to interactive computer systems design. Those applications have taken different forms and have evolved over time along several concurrent streams of work: from prescriptive models of the systems design life cycle, including the role of prototyping, strategies for placing evaluation at the core of the design process and providing methods to support the interaction designer, to descriptive models of human cognitive processes including creative design and strategic knowledge.
Interest in the field of creativity and the role of computer technology in creative tasks has recently increased. A number of national initiatives have been set in motion in the UK, beginning in December 1996 with the Initiative for National Action on Creative Technologies. The principal ones at the time of writing are as follows:

- the Creative Media Initiative of the Technology Foresight Programme from the UK Department of Trade and Industry.


- People and Computers: The Human Factors Programme of the Engineering and Physical Sciences Research Council workshops on initiatives in creative media.

Thus, the author's involvement in creativity research and computer support is proving to be timely. Amongst the recent activities is an ongoing initiative which brings together creative people in design, architecture, music and the visual arts with computer technologists and scientists, for which the author is the technical programme chair. The International Symposium, Creativity and Cognition 1996, marked a consolidation of the first event of 1993 (Candy and Edmonds, 1993, 1996). In the 1996 symposium, the question of how to characterise creativity continued to engage researchers. However, there were noticeable shifts in emphasis from the 1993 programme. One such change was the increasing emphasis on how to address the needs of the human creative process, as distinct from describing and modelling it. More attention is being paid to the process elements of human tasks, skills and behaviour rather than emulating the products of creative work as indications of creative behaviour in the computer.

Attention to human creative processes brings with it a concern for the relationship between the constituent elements and, in addition, the dynamics of learning and the role of knowledge in creativity are brought into the frame of reference. For the most part, the advances in computer system design over the last decade have been such that many of the initial barriers to acceptability have been overcome with respect to applications such as wordprocessors, desk top publishing, drawing packages, spreadsheets and simple databases, and, now electronic mail systems and browsers for searching for information on the World Wide Web. However, the goal of delivering systems that match the needs of
different kinds of users in different domains is still limited. It could be argued that, from the point of view of addressing the needs of the creative knowledge worker, the materials are now available and some of the building blocks are in place, but the brief has yet to be specified and communicated to the architects and designers.

The thesis begins with an overview of the main themes and outcomes of the research which have been developed in a number of publications over time, the main ones of which have been included in the thesis. The papers have been divided into three parts: themes and outcomes, methodology and case studies.

**Part One: Themes and Outcomes**

In chapters two and three, research into creativity and knowledge work and how it might inform our understanding of computer support for creative work is described. Five papers are presented which establish the core themes of creativity, knowledge work and computer support for those aspects of human activities. The outcomes are presented as models and exemplars. A criteria-based modelling approach to interactive systems design is presented which has evolved from earlier models that represent key elements of creativity and knowledge work. The criteria for interaction design were derived from the results of studies in a number of domains of expertise. The extension of the criteria-based approach to the modelling of creative knowledge work is described. A model of creative knowledge work is proposed and categories of criteria identified. Examples of the application of the criteria to three demonstrator computer systems are presented.

**Part Two: Methodology**

In chapters four and five, the motivation, methods and procedures that make up the methodology are presented. Each paper selected for this part of the thesis represents a development stage in the mode of empirical investigation that led to the author's main research outcomes described in Part One. From those starting points, the approach was developed further and gave rise to a set of outcomes that have been applied and tested in real use. These are illustrated in the papers selected.
Part Three: Case Studies

Four of the author’s case studies are presented in Chapters 6 to 9. The studies were carried out in industry/academic collaborative projects where the practical needs of real world development contexts figured highly. The investigations of people, their tasks, work practices and the characteristics of creative knowledge work were carried out in a number of domains: engineering estimation, product design, speech science, engineering design management. The results were considered in the light of other comparable studies.

1.1 Themes and Outcomes

In Chapters 2 and 3, five papers are presented which establish the core themes of creativity, knowledge work and interactive computer system support for those aspects of human activities. The first two areas are combined in a particular way to support the third. The papers have been selected on the basis of how well they represent the main outcomes arising from the author’s work. The journey from prescribing system requirements and design goals to framing the system design process in terms of evaluation criteria may be traced through the papers presented. Underpinning those outcomes are the empirical studies presented in Part 3 and the associated research into comparable work.

The concept of the user as a creative knowledge worker is proposed with relevance to different domains of expertise. Creative Knowledge Work is characterised by the generation of new concepts or artefacts and is to be distinguished from creativity in everyday life (see Fischer, 1993 on creativity in the context of social activities). Developing ideas and making artefacts by any individual may indeed be innovative in the mind of that person and, from that perspective, can be thought of as creative, but the outcomes from such creative acts are not usually available to the wider community. For agreed definitions of creativity to exist, the outcomes become established over a longer term by a process of evaluation and comparison with other ideas and objects (Boden, 1990).

An understanding of creativity and the role of knowledge work may be brought to bear in the drive to improve the design of interactive computer systems. Whilst considerable research is taking place into issues such as improving data transfer, usability and efficiency in task execution, there is a need to look at more fundamental questions about the roles
computers have the potential to play in empowering creative design knowledge workers as users of such systems. This has implications for the methods and techniques that are applied in interactive system design as discussed in Chapter 3.

Chapter 2 is concerned with research into creativity and knowledge work and how the results can be applied to informing our understanding of computer support for creative work. Three papers are presented which establish the core themes of creativity, knowledge work and computer support for those aspects of human activities.

In Chapter 2.1, the paper draws together a number of themes arising from the author's earlier case studies: for example, how creativity has been investigated, whether creativity can be learned and the implications for creativity with computers. Aspects of creativity research are explored and two case studies by the author are described in brief. The studies, which provide examples of creative work based upon the acknowledged creative quality of the outcomes, are presented in full in Chapters 7 and 8. From both source studies, the discussion of the implications of knowledge intensive work, including visualisation and collaboration, for computer support were drawn. The paper illustrates two ways of investigating creative work and the different modelling outcomes that were derived in the form of a process model and a cognition model. The origins and derivation of the process model of interactive knowledge work is described in more detail in Chapter 2.2. A prototype demonstrator knowledge support system is cited as an example of how domain knowledge may be used to provide support as the concept design solution seeking proceeds. This Knowledge Support System (KSS) is exemplified further in Chapter 2.3.

In Chapter 2.2, the process modelling is explained further and the explicit link with setting requirements for system support design made. The work which gave rise to these concepts was a foundation study which had a significant influence on the approach to identifying and defining the requirements of computer support. The objective was to understand the creative aspects of knowledge work and to draw from that the constraints and opportunities for human-computer interaction. A study of a speech scientist using SKI, a knowledge support system designed and developed at LUTC HI, is described. The process
model of knowledge interaction is decomposed into its constituent parts and how it can be used to specify HCI requirements for computer support illustrated. This process model has been revisited and extended into the model of creative knowledge work described in chapter 3.2.

The critical role of knowledge in creativity was first identified in the Speech Science study (2.2, 7.1) and pursued in more detail in the first Lotus Study (Chapter 8). From the design modelling research that informed this work, the importance of the different categories of knowledge that are applied in design was identified. In addition, the related issue of how the knowledge of the team impacts upon the design process was taken up. The investigation into vehicle engineering management was a contributory source to the outcomes of this work which have been partially documented in Chapter 9.1. However, it was the prototype system described in Chapter 2.3 that provided an opportunity to explore the role of knowledge in the system and was a step forward towards developing an integrated creativity support environment.

In chapter 2.3, the paper addresses the question of how to support the designer with appropriate knowledge during conceptual design. It begins with a discussion of knowledge-based support for product design and is followed by a scenario account of the use of a knowledge support system. The system was constructed using the FOCUS architecture and tools described in Chapter 5.2. The LUTCHI Vehicle Packager Knowledge Support System was designed to demonstrate methods that aid designers at the conceptual stage of the design process. Graphical interactive techniques enable the designer to interact with the domain knowledge in specific concept vehicle design terms (Parks and Edmonds, 1997). The results of user evaluation gave rise to the development of a multi-user design support environment for collaborative team work in the MULTIK and ROPA projects (MULTIK, 1997, ROPA, 1997). These demonstrator systems were developed using the criteria modelling framework. Examples of those aspects which meet a set of criteria defined from the case study results are presented in chapter 3.2.
Chapter 3 builds upon the research into creativity, knowledge work and computer support that is reported in Chapter 2. It introduces a proposal for giving evaluation a primary driving role in the computer systems life cycle. In order to carry out evaluation in terms of whether the system meets the cognitive requirements of the creative knowledge worker, it is necessary to specify what those criteria are, and to which parts of the system design they should be applied. The criteria that are cited are drawn from a larger set which have been identified from empirical studies (Appendix 3). It is recognised that the context of the users, their tasks and technical and physical environments must be considered when designing any computer support environment. By applying criteria of the kind defined in 3.1 and 3.2, the attention is focused upon the creative and knowledge intensive aspects of the users' needs. Thus, the characteristics of the creative knowledge worker are relevant to any analysis of the domain that may be carried out as part of the interactive systems design process.

In Chapter 3.1, the paper presents the approach to criteria-based modelling for interactive computer system design. The criteria-based model expresses criteria that may be used to evaluate the design as opposed to task modelling, a representation form from which one might hope to deduce the design. One aim of adopting the criteria-based modelling approach is to re-orientate the way we look at deriving requirements for computer support for creativity. The modelling approach described is based upon the notion of the boundary condition. This involves considering the widest possible range of domain characteristics in order to identify the boundary conditions of the design space within which designer of the computer system needs to operate. These conditions enable the derivation of a set of criteria against which the system design may be evaluated. This paper was the springboard for the criteria model for creative knowledge work described in 3.2.

In Chapter 3.2, the criteria modelling approach is extended and applied to computer support for creative knowledge work. In the model of creative knowledge work described, three main Activities are represented: Exploration, Generation and Evaluation. The Activities are combined with a set of Contributors which feed into the exploratory and evaluation activities and arise out of the generative activities. For each element of the Creative Knowledge Work Model, a set of criteria were identified. The criteria are applied to three levels of system
requirements and examples of applications that meet selected criteria for creative knowledge work support are summarised.

The ongoing work is concerned with applying the criteria to existing applications. A further problem to be addressed is integrating the approach into strategies and methods for project management and development.

1.2 Methodology

In Part 2, Chapters 4 and 5, four papers which present the motivation, methods and procedures that comprise the methodology are presented. Each paper selected for this part of the thesis represents a development stage in the mode of empirical investigation that led to the author’s main research outcomes described in Part One. From those starting points, the approach was developed further and gave rise to a set of outcomes that have been applied and tested in real use. These are illustrated in the papers selected which, although they arise from a diverse set of projects, reflect common concerns.

In Chapters 4 and 5, proposals concerning the computer system design and the role of the systems designer in resolving conflicts are made. Design may be seen as a coordinating activity where the resolution of conflicts is a key task for the designer. The solutions may be different in quality and type but where they meet the criteria for evaluation, this guides the precise choice. This implies that the solution space might be constrained in certain ways but at the same time allow the designer freedom to choose from a range of solutions provided they meet the evaluation criteria. This is an issue which underpins the criteria modelling approach proposed in Chapter 3.

The overview of Part 2 is structured as follows:

Motivations for the research: the issues and factors which underpin the research agenda.

Research Approach: the philosophical position with respect to scientific method.

Outcomes: concepts, models, requirements, criteria, strategies based on findings.
1.2.1 Motivations for the Research

In the early 1980s, before microcomputers, as they were then referred to, were in widespread use, the idea of designing the computer system from a user perspective was in its infancy. The initial field of research in education, described in 4.1, was motivated by certain key concerns which helped to shape the research and design philosophy outlined in 4.2 and were later to be extended to business computing in two large collaborative projects described in 5.1.

The key driving issues behind the work described in these papers were as follows:

- **User-Centred Design**: how to influence the design of computers systems to the advantage of the end user's tasks and work design.

- **Strategies and Methods for Computer System Design and Development**: how to structure and support the total system life cycle process.

- **End User Modifiability**: how to address the existing needs of end users and make computers adaptable to the inevitable changes that will occur as a result of experience.

- **Users as Domain Experts**: how to address the particular requirements of professional people, who, whilst they might not be expert with computer technology, are expert in their own fields and are unlikely to accept its introduction uncritically.

- **User-Centred Design**

  The scope of the research work undertaken has been the relationship between the introduction of computers, the perception of users and the conditions for genuine change in practice. The applicability of a user centred approach across different domains began by bringing the user-centered perspective developed in an educational context (see 4.1) into the business development environment (see 5.1). The first paper can be seen as a springboard for the approach which was applied in other industrial domains (see also 5.2).

- **Strategies and Methods for Computer System Design and Development**

  The way in which successful products are achieved is understood only in a limited way. In the case of software products, the emphasis at this time is on the identification of explicit, often formal, strategies that can be employed. But, computer systems design is also a team-
based co-ordinating activity in which the approach to evaluation and prototyping as vehicles for increasing user involvement and helping re-design the system is an important part of any satisfactory strategy. In combination, user evaluation and the deployment of the user interface prototype as a vehicle for clarifying user requirements, provide practical mechanisms for assessing the appropriateness of the whole system design. This approach was developed in FOCUS, a major European project in which the needs of the user companies drove the direction of the outcomes which have since been applied in another industrial context. The lessons from applying the FOCUS strategy and methods, described in 5.1, paved the way for the criteria modelling approach presented in 3.1 and 3.2.

- **End User Modifiability**

A Knowledge Support System (KSS) is one that enables *direct* interaction between domain expert users and the system and is designed to address the problems of capturing expert knowledge (see 2.1 and 7.1). The KSS under consideration is distinctive from typical computer support systems in two particular respects: firstly, the system is designed to support domain tasks of a higher level order and, secondly, it must be flexible enough to handle the dynamic nature of the user knowledge and the emerging requirements that accompany its evolution as the expert progressively refines and extends his or her knowledge. The KSS concept was extended into a scientific workbench called the Speech Knowledge Support System (SKSS) which provides the domain expert access to several system modules and existing applications that are integrated at the user interface for maximum consistency and fluency (Agarwal & Candy, 1992). The system was implemented using the FOCUS Architecture and toolset described in 5.2.

- **Users as Domain Experts**

The Professional User: the user groups included in the cases studies were teachers, estimators/ engineers, engineering management, chemists, statisticians, speech scientist/phonetician, product designer/engineer, vehicle engineer and computer system developers. Observations of the work practice of such users showed that people with specialized expertise are unlikely to accept innovation uncritically. The reasons for this vary from situation to situation but the overriding issue is whether they can see an immediate benefit or some additional value in adopting a change of practice. If the computer use does not provide this, no amount of special effects will compensate. In addition, the
work environment is often critical in terms of time constraints and, in highly demanding situations, any additions to the workload or existing complexity will reduce productivity. If the computer use requires more effort, there must be significant payback to make it acceptable. This assumes that such users have discretion in whether they adopt its use or not.

All of the above issues are relevant to current thinking and continue to underpin the criteria-based modelling approach proposed in Chapter 3 of this thesis. From those early starting points, the methodology has been developed along particular lines of thought and has given rise to a set of outcomes that have been applied and tested in real use. These are illustrated in the papers selected for Part 2 in relation to life cycle models and systems development strategies and methods.

1.2.2 Research Approach

The philosophical position with respect to scientific method is presented in Chapter 4. In 4.2, it is argued that the scientific paradigm, based upon the rationalist tradition of enquiry, has dominated computer system research and development, resulting in a preoccupation with the technology itself, its performance and formal characteristics. When the human user of the system is brought into the scope of concern, the tendency has been to import the same approach: that is, in its simplest form, to contrive situations where the user and system can be observed and monitored, a limited set of variables can be manipulated and the resulting model can be used to make predictions.

By contrast, the Action Research method described in 4.1, aims to identify existing assumptions in practice and then develop new strategies in the light of that knowledge. The term, coined by Lewin (1947), referred to a process which embodied the integration of practice and research. It has affinities with the Soft Systems Methodology (SSM) Checkland (1984) which attempts to move away from describing the real-world situation to taking action in the situation.
An important dimension of Action Research is the role of participation and it is this that links it to the concept of 'reflection-in action' (Argyris and Schon, 1974) where the practitioner's knowledge is embedded consciously in action. The role of participative research is central to the early computer-aided work described in 4.1 and is developed further in 4.2. Since the author's early work, both action research and participative research and design have gained a place in the Human-Computer Interaction and Design Research communities and are now widely referred to when investigating user-system behaviour. Schön’s theory of Reflective Practice (Schön, 1983) is explored in depth by Dorst (1997) in the context of design research.

A particular feature of research investigation into users, their tasks and computer use is the collection of large quantities of qualitative data. The quest for the "rich picture" brings with it significant problems with respect to reduction and analysis of that data. The methods employed in the studies presented in the thesis, included the use of a number of data collection techniques: observation checklists, questionnaires, field diaries, interviews, system logging and playback, audio and video recordings. The data analysis provided a variety of outcomes including immediate feedback to be translated into modifications and directional changes in the system design. This is a means of making improvements in the current context rather than at a later time. In this respect, the research approach differs from more traditional experimental research which is directed towards seeking generally applicable results.

1.2.3 Outcomes

The 'action-based' outcome of the research is key characteristic of the total methodology presented in the thesis. In the case studies reported, the findings gave rise to outcomes which take the form of models, requirements, criteria, strategies, methods and techniques. The case studies yielded different kinds of evidence about creative knowledge work which were used to test existing models. These were then modified, or new models were derived in the light of that evidence.

The models, in turn, informed the requirements for creativity support systems in which knowledge evaluation and extension, visualisation and collaboration are proposed as
essential ingredients of creative knowledge work (see 2.1). However, whilst requirements lists provide a necessary starting point, they do not provide sufficient guidance to the interactive systems design process itself. The work undertaken in the FOCUS project, described in 5.2, in developing both strategies and methods for supporting the computer system development life cycle, provided insights into the difficulties imposed by making the support methods too prescriptive. This was also especially evident in the application of the Hierarchical Task Analysis (HTA) method (Shepherd, 1989) to the design of the Speech Knowledge Support System (SKSS) (Aggarwal and Candy, 1992). It became clear that a framework that provided structure to the process was needed but also that flexibility in the solution generation was necessary from the system designers’ point of view. From this experience and other investigations of creative work, the criteria modelling approach that enabled an evaluation framework to be put in place, arose (See 3.1 and 3.2).

The outcomes presented in Part 1 of the thesis are expressed as models and criteria for interactive systems design. Those that are presented in Part 2 represent attempts to establish the strategies and methods of a user-centred view of systems design, as distinct from a system-centred approach. There is a need to provide a flexible framework for interactive systems design within which a variety of methods and techniques can be applied where and when appropriate. That framework must be responsive to the emerging situation and yet provide tangible and immediate results that inform the particular case immediately. Deriving knowledge that is immediately applicable requires different techniques, particularly if it has to address multiple issues in context, which imply a different granularity of research problems to be addressed (see 4.2 and 5.2).

In the model outcome described in 5.1, the identification and refinement of requirements is seen to continue well into the system development process. A prototype stand alone user interface to assist with the early requirements analysis was used and described in this 1988 paper. It is an approach that is now firmly established in HCI methods for user-centred design (Preece et al, 1994). The strategy and methods described in Chapter 5.2 have been applied in industrial and commercial companies and thus the applicability and usability have been demonstrated. The next section provides a summary of case study findings that gave rise to the outcomes.
1.3. Case Studies

In Part 3, four of the author’s case studies are presented as Chapters 6 to 9. The studies were carried out in industry/academic collaborative projects where the practical needs of real world development contexts figured highly. The investigations of people, their tasks, work practices and the characteristics of creative knowledge work were carried out in a number of domains: engineering estimation, product design, speech science, engineering design management. The findings were considered in the light of other comparable studies.

1.3.1 Definition and Purpose

A case study is an empirical investigation of a specific set of events within a real-life context in which a number of factors are considered as evidence (Craig Smith, 1990, Bruce, 1993). The units that are studied and analysed may range from individual studies of outstanding designers to histories of innovative corporate culture. It has been used extensively in action research and ethnography. It is most applicable when events are not amenable to control by the investigator and when the questions posed are open ended and multi-factored. Asking questions about why and how something took place is undertaken in order to understand the meanings of the specific instance at hand. The explanations are based upon observations of existings events or recoveries of past events. The findings from such studies do not directly generalise although it is common to compare results with other similar studies. A common use of the case study is to generate hypotheses about a wide range of events which may then be studied in single variable controlled conditions using traditional experimental methods.

Case studies provide evidence about how designing takes place in real world contexts. Empirical testing involves testing knowledge claims within a context (Argyris et al, 1985). The results of observations have a bearing upon the acceptance or rejection of a theory. The connection between the theory and the results of observation should be explicit such that people can agree them. In research into human activities, controlled laboratory conditions are not achievable without sacrificing the context that gives them meaning. Instead, there are richer layers of meanings which are imposed by the social context and which are relevant to the description and interpretation of what is happening. When studies of design and creativity are carried out, the context is an important consideration. The
studies may be carried out at different levels of the problem scope and examine this from different orientations from cognitive dimensions to social dynamics and design strategies.

1.3.2 Boundary Case Method

In statistically-based experimental science, the unusual cases that are single instances and fall outside the norm are inclined to be discarded as resulting from some error or other. Multiple instances in the same context are reinforced and, therefore, more confidence can be ascribed to the results. In this approach, the characteristics of the working practices of (say) the outstanding designer, who operates in a highly individualistic way, are likely to play a very small part in the analysis of the data.

Popper's proposal was to see science as an activity which spent considerable effort in looking for counter examples (Popper, 1969). From this point of view, a single instance that does not match the theoretical model of events, far from being discarded, can be vital information. An important aspect of this is in the consideration of an observation in a specific context.

- Repeated changes of context with the same phenomenon occurring leads to a belief in a 'generic' conclusion.
- If a change of context leads to no instances of the same phenomenon, then it can be assumed to be related to or be dependent upon the context.

As an example, consider the case of the public/private space requirements that are relevant to design support for teams and individuals. It is necessary to resolve when and where to allocate these potentially conflicting features of the system and to make them available to the users as appropriate to a given context. The interactive computer system designer has to know about conflicting requirements and other information about the users and task/domain context in order to resolve them. From the interaction designer's point of view, a single conflicting instance may be seen as a signal that flexibility in the requirement exists and must be addressed. A model of the user domain may have entries that are based on significant instances that are important in generating the flexibility of application of the system.
Another way of considering this issue is to say that considering the boundary conditions provides insight into what is significant about the users, tasks and the work context. For example, in perception, an understanding of illusion can be most informative in revealing the core issues (Gregory, 1974). The creative knowledge worker may be seen as existing at the boundary of the space of users in general.

In respect of informing computer support design, evidence about the designerly behaviour and working practices of outstanding designers provides the following:-

- Given that they are likely to be demanding of the conditions in which they carry out their work in respect of their tools, materials, resources, personal strategies, this may be used to test the limits of what is acceptable and possible for computer support.

- They are known to be ready to disregard conventional wisdom and to hold opinions on a wide range of topics and, therefore, may be expected to provide a rich source of new ideas and alternative perspectives on any issue.

- Exceptional cases are likely to provide a sharper focus on the differences between people's working practices as distinct from the commonalities. This is particularly important if the boundary conditions of the design space are to be understood.

- The results may provide models of good practice.

- Outstanding designers are capable of presenting their ideas and insights in a fresh, but highly organised manner and the result is often far more accessible than a single instance would suppose. Many examples do not always lead to greater value.

Visser draws a distinction between what can be concluded from evidence about so-called novice designers and experts (Visser, 1994). It can equally be claimed that the study of exceptional people working in their normal environments is likely to yield quite different results from investigating the practice of trainee designers in laboratory conditions. The case for the study of outstanding designers rests upon the reasons given above. These factors may be used to influence the direction of computer support system design.
issues. From case studies of creative people at work with and without computers, examples from the author's research are described in the next section.

1.3.3 The Studies
The case studies, presented in Part 3 Chapters 6 to 9, provide evidence about the characteristics of outstanding people that underpin the concept of Creative Knowledge Work referred to in section 1.1 above. The studies were carried out with experts who were not only experienced in their field, but who demonstrated exceptional ability to come up with innovative ideas or products. The findings suggested that, in these cases, the expert's knowledge is undergoing continuous reflection and renewal. Creative knowledge workers were seen to exhibit multi-dimensional goals and to formulate problems requiring the generation and evaluation of entirely new solutions. In this type of work, the tasks are not performed in a manner that conforms to a predictable set of operations and sequences.

The studies presented in Chapters 6.1 and 7.1 were of the use of interactive knowledge-based systems which were intended to be 'co-operative'. Thus, a requirement for human-computer collaboration was identified at the outset in both cases and the studies considered what that meant from a user point of view. In the case of the expert system described in 5.1 and 6.1, it was intended to provide a cost estimate on the basis of a specification provided by the user. Where the output from the system (the operation layout) was incomplete, the user was expected to carry out its completion by adding the missing information. There was no provision for amending and adding to the existing rule base, nor was there a capability for the knowledge base to learn from experience. In this case, the end users did not consider that the system provided sufficient expertise to be able to generate an adequate result and were not convinced that their own work was enhanced by its use.

In the case of the Speech Knowledge System (SKS), the user was provided with graphical interaction tools to inspect visual source data and to test and modify the rules as the work progressed. In this case, the knowledge base underwent continuous testing by the user and where the rules were found to be effective, they were included. Whilst it was clear that the user found the system very effective in carrying out the work, it was not considered to be truly a 'cooperative' relationship in the sense that both parties contributed equally, albeit
differently, to the process. In both cases, the process of co-operation between user and computer system (i.e. the operational features) was considered to be a significant issue in itself, as distinct from the functions made available and the accuracy of the knowledge generated by the system.

A summary of findings from each study appears in sections 1.3.3.1 to 1.3.3.4 below.

### 1.3.3.1 A Study of the Introduction of an Expert System

The Study presented in Chapter 6.1 took place in an estimating office in an engineering company. The application was an expert system for the bid estimation of machining costs prior to tender. The end users were experienced cost estimators, the indirect users were the estimating department management and the data processing department. The following topics were investigated by the study:

- user work task scope
- user perceptions of system use
- accuracy, organisation and presentation of knowledge;
- appropriateness for the task
- dialogue design; style of interaction
- functions and interaction routes
- screen design and layout

Selected findings are expressed as criteria for evaluation below.

- **Support for User Task Methods**

  The system should:
  
  - provide on-line access to scheme drawings relevant to the ongoing task
  - provide support to drawing analysis
  - provide assistance which broadly conforms to a user's existing methods
  - allow the expert to make decisions *during* the consultation process which is then taken account of in the subsequent advice given by the system.
- provide a problem solving approach modelled on how people co-operate in the task domain.
- provide support for on-line information during the development of the task.
- reduce time taken to complete the whole task using the system
- provide user management with a degree of standardisation of the end result.

**Interaction Style**

The system should:

- not constrain the user to a fixed order of interaction with domain task
- allow the use to go backward as well as forwards in the modification and completion of the advice provided by the expert system
- allow the user to interact with the system output in whatever order is required at the time and to take opportunistic steps to achieve a given goal.
- allow the user to revert to a previous stage and modify the decisions made.

**Performance and Reliability**

The system should:

- provide accurate and comprehensive knowledge in the advice to the user.
- be fast and reliable at all times.

**1.3.3.2 A Study of an Expert's Practice**

In Study 2, (7.1) key aspects of a scientist's working practice were identified. Cognitive processes were identified and expressed as goals and intentions, procedures, extensions to and influences on knowledge and its application. The system provided a means to formalise and encapsulate expert knowledge. End user knowledge manipulation techniques enabled a direct and accessible method for testing existing knowledge. The results of the study contributed to a framework for system design which included establishing a number of design goals which are expressed as criteria as follows:

**Interaction Style**

The system should:
enable an holistic view of visual data that includes multiple views.
- enable simultaneous access to all forms of relevant data and rule input methods
- allow the user to change the sequence of actions at any time
- provide the user with a flexible and exploratory mode of interaction
- should enable rapid feedback to user about rules applied in knowledge base and give support to evaluation that includes negative and positive results
- should enable knowledge base to handle partial information.

1.3.3.3 A Study of a Creative Designer

In Study 3 (8.1 and 8.2), the findings suggested that, in this case, creative design is comprised of a diverse set of activities in which there is interplay between different factors. The findings suggested that creativity in design involves the following:

- total immersion in generating ideas.
- a role for sketching in developing, refining and transforming ideas
- adopting an holistic or systems perspective to scope the problem fully
- identifying questions to provide a systematic decomposition of the problem
- ability to make analogies with other fields and product areas
- learning new methods in response to need
- developing idiosyncratic strategies
- using deep historical and practical domain knowledge
- keeping abreast of new events in the field.
- paying close attention to other experts.

*Knowledge Development*

During the design process the designer is constantly learning and gathering knowledge from a wide range of sources. Knowledge development evolves over many years before breakthrough ideas emerge. This evolution does not necessarily take place within the closed set of rules that represent conventional wisdom. The creative step is one in which a
1.3.4 Conclusions

The principal aim of the research undertaken by the author has been to identify the characteristics of creative knowledge work across different domains of expertise with a view to informing the design of interactive computer support systems. The case studies that have been carried out have investigated the cognitive aspects of the working practices of highly expert people whose work has given rise to new knowledge of one kind or another. The findings from those studies were considered in relation to other similar studies of experts and, from that work, such professional people are here characterised as 'creative knowledge workers'. It is suggested that existing computer systems design methods do not specifically address the characteristics of such users. In the last decade, the rapid changes in user-centred computer technology have provided significant improvements to the resources and facilities available to the everyday user. However, there are important challenges that remain to be addressed in order to meet the needs of creative knowledge workers. To that end, the thesis proposes an approach to computer systems design and development that directly links the requirements definition and design specification to the application of evaluation criteria based upon the characteristics of the cognitive style and working practices of creative knowledge workers.

References


conventional rule is broken. The designer is a knowledge worker involved in creative work that is not easily characterised by formal procedures. Computer support, therefore, is not most easily provided by implementing well defined processes. What is required is support that is flexible and usable in the context of the attributes of the thinking processes that the user is engaged in.

Studies of creative people have shown that are able to make associations between different categories of ideas and to be willing to take more risks. The design problem is tackled on a broad front and the candidate solutions are defined as a part of a whole system rather than decomposed into detailed parts. Alternative solutions are explored by examining a range of options and trying where possible to keep a number of channels open. New ideas often arise by considering ideas or designs in different domains. The advantage of being able to envisage solutions to problems drawn from outside the particular domain is that the constraints are not fixed. These were all characteristics of the people studied by the author and by other researchers. The use of in-depth domain knowledge and access to state-of-the-art expertise was critical to creativity. The findings of this study were compared with other studies (Maccoby, 1991, Roy, 1993, McNeill and Edmonds, 1994, Cross and Cross, 1996).

1.3.3.4 A Study of Engineering Design Knowledge

In the final Study 4 (9.1), an investigation in vehicle engineering design was carried out. The outcome is a set of findings that are relevant to design modelling and interactive computer systems design. Examples of strategic knowledge within the overall design process were identified, some of which are listed below.

- When to apply constraints
- Solution-Led Strategies Vs Problem-Led Strategies
- Use of Knowledge and Experience in Solution Selection
- Individual Designer Strategy

The study gave rise to questions about the appropriateness and effectiveness of the representations used. Three types of models were described and the purpose to which they could be put as follows:

- a model of strategic knowledge for supporting interactive system design
- a model for representing strategic knowledge computationally
- a model of strategic knowledge in engineering design


2.1 COMPUTERS AND CREATIVITY SUPPORT: KNOWLEDGE, VISUALISATION AND COLLABORATION


This paper is concerned with research into creativity and how it might inform our understanding of knowledge-based computer support. Aspects of creativity research are explored and two case studies by the author are described. The implications of knowledge intensive work including visualisation and collaboration for computer support are discussed. Finally, future research directions that combine the aims, objectives and techniques of both the Artificial Intelligence and Human-Computer Interaction communities are outlined.

Keywords: Creativity, Knowledge, Computer Support

2.1.1 Introduction

The strengths and weakness of current computer systems may be judged according to whether they meet the user's skills, needs and expectations of them within a given context. Users also have different levels of competence and task goals and these factors impose particular requirements upon the design of such systems. Most people do not require any detailed knowledge of the computers in their washing machines or cameras and experience tightly constrained forms of interaction with them. Interacting with a vehicle monitoring computer, for example, is not usually considered to be a desirable function. You may ask it for information about temperature or petrol consumption but, in most cases, when the symbol for a routine check appears, whilst you may be aware that something is taking place, you cannot ask for a status report on your vehicle. The service engineer, on the other hand, needs to have greater access to the information gathered during a drive, which may be very important to how effectively he can carry out his work. In these instances, the degree of interaction with the computer is circumscribed by its perceived primary function of monitoring and fault detection and, in effect, its role is primarily a one-way transfer of
information. In respect of route navigation systems, the degree of interactivity is different again and the introduction of such systems raises an entirely new set of issues about task support.

2.1.1.1 Creativity Support?

Whilst acknowledging that there have been vast improvements in computer systems for personal and business use in the last decade, there remain important aspects of the design that constrain the users to prescribed ways of carrying out their work rather than support them in the most appropriate way. Access to the full computational power of a system may be obtained by learning the programming languages that drive them; but, for the ordinary non-computing specialist user, these languages are arcane and thus, the extent of that power remains highly circumscribed. In spite of significant developments in the role of the computer for creative tasks, we have yet to marry ease-of-use with full access to the computer's potential. Certain applications are, nevertheless, more successful in satisfying user needs and providing appropriate task match than others. Spreadsheets and some CAD applications have been shown to combine appropriate task notations with usability [1]. These applications are being used by artists, graphic designers and educationalists who are experts in their own fields, although not in conventional computer programming.

A question arises that if, indeed, computer systems are becoming more accessible and usable, is there any reason to doubt that they will eventually meet the requirements for creativity support through the normal evolutionary process of adding functionality and usability to existing applications? To some extent, the answer lies both in the nature of creativity itself and in the underlying attributes of computer system design. In creative work, the generation of new concepts by (say) the designer, which might include, for example, new objects formed from old ones, cannot be easily handled in current computer systems. In creative work, knowledge about domain entities, whether they take the form of visual shapes, objects, parts, complex products or textual or analytical/numerical data, plays an essential role in the development of the task in hand. In current computer systems the objects, structures and procedures that are provided by the system are usually
embedded (even hardwired) in such a way as to make them inflexible and inaccessible to change. Thus, for example, if a new shape is drawn by the user, the information about its composition, dimensions, etc. that may be necessary to manipulate it, compare and evaluate it, is not available within the system. More problematic is the handling of shapes that have emerged in the mind of the designer as he or she observes those set down and then wishes to manipulate 'new' shapes. This is an important and growing area of research. For further discussion about the nature of emergence and why computer support for creativity must take account of it see [2-4].

The above example is only one such illustration of how we need to have greater understanding of creativity in order to inform the design of support systems. Whilst considerable research is taking place into issues such as improving data transfer, usability and efficiency in task execution, there is a need to look at more fundamental questions about the roles computers have the potential to play and what more can be achieved in the direction of empowering users. In particular, there are issues arising from empirical studies about the role of knowledge, visualisation and collaboration in the creative process that need to be addressed.

2.1.2 Systematic Approaches to Creativity Research

The systematic study of creativity is largely a post World War II development within the discipline of psychology [5]. Academic interest prior to that time was the prerogative of the historians of art and science. Doing creative things was, of course, the business of the artists, writers and composers who were not expected to provide reliable evidence about the nature of creativity itself even though it was their primary occupation. The elusive nature of creativity stems largely from the assumption that creativity is not open to experimentation in the laboratory and the retrospective accounts and anecdotal evidence available was unreliable and certainly not refutable using scientific methods.

Early characterisations of creativity provided points of departure for the more systematic approaches. Those approaches used different frames of reference, from the psycho-
analytical and psycho-pathological ones to Gestalt, trait formation and associative kinds [6]. Much valuable pioneering research took place into areas such as the creative personality [7, 8, 9] and the relationship between creativity and intelligence [10, 11]. Albert [12] documents the history of previous work in Social Psychology into genius, creativity and giftedness. He notes that there has been less interest in identifying what creativity is and who are creative people than one might have supposed.

Attention to the creative process itself drew upon reports of highly creative people (e.g. Poincaré's own account of his mathematical discoveries [13]. Work on defining the criteria for a creative product, a complementary stream of activity, also took place [14]. Sprecher proposed the notion that a truly creative product or idea has the characteristic of being itself creative in the sense that it generates additional creative activity [15].

2.1.2.1 Can Creativity be Learned?

The scientific research goal of trying to define the characteristics of creativity began to be accompanied, in certain groups, by a very different goal, that of trying to develop creativity as an educational process. Whether or not, those ways of thinking that were found in creative people could be nurtured in the population at large was an issue that was much debated. Methods for investigating the stages whereby a creative idea is generated and turned into a creative outcome were devised and applied. For example, Parnes' brain-storm programme used a five step procedure: fact finding, problem finding, idea finding, solution finding and acceptance finding that was applied in the Creative Problem Solving Institute at Buffalo [16].

The issue of whether or not creativity can be understood or characterised in a reliable way remains a research issue despite the vast quantity of knowledge from the last fifty years of research. The development of a coherent picture of the multi-dimensional aspects of creativity, of which a small number are referred to above, is yet to be achieved. The issue as to whether creative thinking or behaviour can be learned, or encouraged, invites a set of
questions that are different again from those posed by the quest to understand its characteristics.

In seeking to promote creativity, does this mean that we must base the approaches used upon tried and tested results about how creativity takes place? If we argue that it is important to adopt creativity supporting approaches that have a sound basis in research evidence, then we need to be very clear as to what kind of evidence is needed and what is reliable. A fundamental problem here is that much of the evidence could be considered to be inconclusive and, in certain cases, contradictory. It might also be the case that some of the evidence does not suit the prevailing wisdom of the time. For example, if we consider the evidence that individuals are better at problem solving than groups using brainstorm techniques [17] should we reconsider the drive for increased team work?

2.1.2.2 A Climate for Creativity?

One area of research, that bears upon the question of whether creativity be learned or influenced in some way, and has received less attention is that of the conditions in which creativity takes place. These conditions might be defined in terms of the environmental (including organisational) factors, and indeed the materials or tools used to achieve the creative outcome. Where experiments into creative thinking have been carried out, the impact of the materials used have proved to be influential in making differences to the outcome [18].

One might hypothesise that the characteristics of any resources, materials, tools or techniques that form a part of the creative work are in themselves critical factors that influence the way it takes place, i.e. the process. It follows that the characteristics of the support environments, whether computer-based or not, ought to be determined on the basis of what we know about the creative process. A further question arises that, even if a support environment is designed to meet the creative process of the creative user, how far does it, in turn, change that process? Creativity is not responsive to deliberate efforts to initiate it because its occurrence is highly unpredictable and is, therefore, resistant to
planning and control. The same might be said for supporting it by computational means. Perhaps in the end, the issue might not be so much how to support creativity or encourage it, but how to avoid impeding it when using computer tools?

2.1.3 Empirical Approaches to Creativity Research

Empirical research into the nature of human creativity takes many different forms, including ethnographic studies, case studies, protocol data studies, controlled experiments and simulation trials using AI techniques. Where a computational perspective applies, there are two main streams of research. The first arises from traditional AI where the goal is to reproduce human creative process or creative products as computer programs see[19-24]. The second is one where the goal is to support or enhance the human creative process and, by implication, enable it to be passed on or learned see [25-27]. In both cases, the need to develop computational models, albeit of different kinds, is implied.

Whilst there are many areas of overlap between the above areas of research, the ultimate goals are different. The development of computational models for exploring the nature of creativity by reproducing that behaviour in computer programs is most often dedicated to the derivation of fundamental scientific knowledge [28]. By contrast, the development of creativity support has, on the whole, a more immediate and pragmatic aim, that of informing the design and implementation of computer support environments for creative work. In each case, the outcomes of the research differ in respect of understanding how that knowledge obtained is to be used. The existence of a program that emulates human creative process has intrinsic value only insofar as how well it may be validated against the original hypothesis. On the other hand, the creativity supporting computer environment must be acceptable to the human beings who use it and also demonstrate to their satisfaction that it does, indeed, support them in the creative aspects of their work.

There are also methodological differences between the two approaches, in particular, the extent or scope of the creative acts that are to be considered. It is possible to research the creative process in its "small acts" such as identifying how incubation takes place by way
of laboratory experiments (see Partridge and Rowe for a review [28]). Alternatively, creativity may be seen in the larger scale, where all issues, including the creative outcome itself, are considered. This all embracing approach is a broad brush first stage from which specific issues may be identified and pursued in more detail. An example of such an issue would be the identification of the important role of perceptual changes in creative insight and the research area of emergent shapes that has arisen as a result.

Weisburg [29], posing a challenge to some fundamental concepts of creativity, has claimed that there are no generally describable creative mechanisms that some people possess and others do not. Evidence that certain people, such as scientists, are not particularly adept at so-called 'divergent thinking' supports the view that creativity really depends upon the domain specific skills being used. However, it should be noted that this work and the critique by Partridge and Rowe arises from an analysis of creativity "in the small". What is missing from this kind of evidence is how creativity takes place in the context of realistic situations outside the confines of the laboratory.

An alternative approach is to examine creative work (as defined by the creative outcomes) in terms of the totality of the person's activities. Creativity looked at this way can be considered to be characteristics (e.g. mental operations such as memory, recognition, intelligence etc.) that are combined in an exceptional way so as to maximise their effectiveness [30]. The traits might be extended to include basic cognitive capabilities, values, motivations and strategies. Another approach, that of deriving theoretical models of the creative process is a different starting point. However, such models do not tell us what we need to know about the general context in which creative work takes place, nor do they give us insight into specific scenarios that tell us how ground-breaking ideas arise. Hence, the need for the case study that encompasses a wide range of elements.

The value of case studies of individual people in giving real-life proportions to the general theories of creativity has been advocated, although it must be said, not carried out extensively. Theories of innovation in technology have either tended to dwell upon the
artefact itself, using an economic or social explanation [31]. The value of case studies of individual creative people in providing depth to our understanding of the total process is discussed below and illustrated by examples in the following section.

2.1.3.1 Case Studies of Creativity

Studies of the creative process "in the large" (i.e. complex factors of the motivations, activities and outcomes as they happen in context) have been carried out in a number of empirical case studies of successful individuals who have demonstrated by the outcomes of their work that they have made a special contribution to innovation.

Maccoby [32] studied prominent designers and engineers whose contribution to their fields was unquestioned by their peers and the world at large. Although they represent a spectrum of different fields and cultures, they exhibit similar ways of thinking and working. Most are "systems thinkers" in the sense that they look for an overall broad scope before moving into specific detail. Masaru Ibuka, co-founder of Sony Corporation and originator of many electronic firsts describes his experience as follows: "I think over the issue I am working on and wait until I get a vision that illuminates its overall nature." This process is a necessary fore-runner to that of discussing with his team how to put the ideas into effect.

This ability to think "holistically" is related to another important feature of creative thinking and one that is vital to the breaking out of conventional ideas and solutions, especially when faced with a dead end or set of apparently unresolvable conflicts: this is an ability (and preference for) working from first principles. Gordon Murray, chief engineer for the Brabham and McLaren Formula one racing teams in the 1970s and 1980s, preferred to keep experience at the back of his mind (he meant specific instances or solutions) and to work from first principles. For example, in designing a wishbone suspension system, he observes:

"it's all too easy...to say, I know all about wishbones, this is how it's going to look because that's what wishbones look like. But if you want to make a step forward,...then
you have to go right back to load path analysis. It is like designing things for the first time, rather than the n-th time." [33].

From the results of empirical studies, certain characteristics of creative design that are relevant to this discussion have been identified. In particular, design is often solution-led, in that early on the designer proposes solutions in order to better understand the problem [34]. Thus previous, prototypical designs are needed and re-use is a significant activity [35]. Secondly, however, it is seen that the iterations resulting from the solution-led approach are largely driven by the perceived constraints upon the design goal. These constraints (be they requirements, legal issues, stylistic preferences or whatever) generate the criteria against which each candidate solution is evaluated and the next iteration initiated.

In both product design and software design, characteristics in common have been identified: in particular, design as an hierarchically organised planned activity versus design as an opportunistically driven mix of top-down and bottom up strategies has been explored in a number of empirical studies see [36-38]. The results indicate that the factors that differentiate between design situations are complexity of the problem domain (e.g. aircraft or vehicle design compared to simple products), whether the problem is ill-defined or well-understood, the experience and skill of the designers and the operational conditions of the total activity in a given organisational context.

From case studies of creative people at work with and without computers, examples from the author's research are described in the next section.

2.1.4 Studies of Creative Process

The studies reported provide examples of creative work based upon the acknowledged creative quality of the outcome of the process itself. Some characteristics of creative cognition that coincide with findings from other studies were identified, in particular,
Maccoby's investigations into high profile engineers, Roy's studies of innovative product development and Cross and Cross's study of Gordon Murray [32, 33, 39].

Two studies are described, one in Speech Science [41,42] and the other in Product Design see [42-44]. In the first instance, the work took place using a knowledge-based support system throughout the process. In the second example, a computer system was not involved at any point in the creative process.

2.1.4.1 An Example from Science
In Science, if creativity is acknowledged, it is usually perceived as the results of investigations, in the H-creative sense [22]. If ideas or concepts are recognised as being original contributions to the field, they have, in effect, been accepted into the canon of scientific knowledge. On the other hand, the creative process, whereby the scientist achieves new insights, is framed, indeed, constrained, by the scientific methods and validation techniques used.

Scientific investigation is an appropriate context of discovery and affords opportunities for studying the creative process. In order to understand creativity in the context of actual scientific research, there is a need for further studies of scientists carrying out their investigations in a real context. In this way, we can gain more insight into the ways ideas are generated, considered and pursued. In scientific work too, it is also necessary to include other aspects of human thinking processes that provide support to scientific exploration such as reflection and insight and the role of personal practice, as differentiated from the application of knowledge derived from reference points in the scientific canon.

Scientific knowledge workers can be provided with powerful methods for studying and evaluating their source material using computer support. Graphical techniques for marking up visual phenomena and expressing knowledge about that data in rule form are available. The need for and use of visualisation of data in the scientific domain is now well established. The scientist may view and manipulate high quality domain specific data such
as aircraft models, molecular structures, brain scans or architectural models. The facilities for changing the parameters, and hence the "view", enables the such knowledge workers to explore the visual data in order to understand and interpret it more effectively.

In the study referred to [40,45], the visualisation of such data is combined with explicit rules about the associated knowledge that has been compiled by the scientist. To enable the scientist to express knowledge to the system in a domain specific language, a bridge between the representation of visual data and the representation of scientific theory in the form of rules was made available. In effect, the visual data analysis and interpretation is combined with methods for capturing formal domain knowledge as rules. The scientist uses graphical interaction techniques to annotate visual data and to specify the knowledge gained from studying the data. Having assimilated the results from the exploration of the visual data and the knowledge base, the scientist can continue to advance existing knowledge further as the analysis and interpretation progresses. The aim is to reduce the constraints upon the scientist's explorations and unpredictable courses of action.

The Speech Knowledge Interface (SKI) system [46] supported rapid graphical interaction with the visual images. The annotations were recorded for later use at the testing stage when the scientist asked for an identification of the annotated speech utterance. The marking of the visual features using colour, blocks and lines appeared to heighten awareness of the finer details of the image. The overlaying of alternative quantitative information about the visual data supported more reliable judgements. The speed, quality and quantity of visual data access was vital as multiple images were required simultaneously and the interchange between them needed to be quick and fluent. Unpredictable features were identified that could not be marked up using the standard techniques and the scientist used an improvised solution. This often led to a need to refine existing rules.

Having evaluated the results of the knowledge base identification, the scientist could then change the rules or add new ones according to the fresh insights which arose during this process. For example, having identified the need to take account of the effects of contextual
variants on the recognition of particular visual features, new rules were created, tested and evaluated about those effects. In total, the process was an iterative one in which ideas were applied, abandoned or refined as the discovery of new insights took place. When a solution, in the form of a refined rule, proved impossible to achieve, the existing rule or rules was typically discarded. It was then that the whole problem was reformulated and an entirely new rule created.

The SKI system contained within it a set of constraints within which the scientist set out to achieve certain goals. The extent to which the scientist moved beyond the boundaries of those constraints and created new concepts can be seen as an indication of the creativity involved [47]. The methods of interaction were designed to enable the knowledge worker to manipulate the visual source data and the knowledge base in a flexible manner.

The fact that the feedback on the knowledge being applied was rapid and immediate was useful but that was not the main advantage. The rules were created and then evaluated against the source data under scrutiny by requesting an identification. This implied that the speech knowledge captured was tested rigorously. Because it represented the current state of the scientist's thinking, the continual process of confirmation or refutation as the results of the identification process were produced was quite powerful. This was a form of support that challenged the scientist's hypotheses and forced a response at the knowledge level not just the surface interaction level.

The study illustrated how, in creative work, exploratory ideas and acts arise during the process and sometimes as side-effects rather than from the explicit objectives being pursued at the time. By its very nature, creativity cannot be described in advance and this makes the modelling task somewhat challenging. In particular, the application of knowledge that is highly expert, distinctive in character and constantly evolving is a feature of the way creative people work.
From the study, a model of the creative process was derived. It represents the key interactions between user and computer system in respect of three areas: Generation and Invention; Exploration and Evaluation and handling Constraints.

The creative process as observed in this study and in others [48] may be described in terms of three main stages which the user moves between: generation and invention, exploration and evaluation and the consideration of constraints.

Figure 1: AProcess Model of Creative Work
Taking each of those stages in turn, we can decompose them and consider the roles of user and computer system more specifically.

* **Exploration and Evaluation** consist of examining the data, evaluating and refining the rules. In this case, the balance between human and computer shifts during the process. Whilst the human examines the data and refines the rules that the computer presents, the data is analysed by the computer according to the existing rules which are, as a consequence, evaluated by the human.

* **Generation and Invention** involves, in addition, moments of insight and the creation of new rules. Both of these are human activities. In particular, any insight obtained is an insight of the human understanding.

The consideration of **Constraints**, on the other hand, involves receiving and clarifying them, revising, and possibly negotiating, the revisions. Here, then, the computer presents the existing constraints, the human revises them and the resources of both are employed in the process of considering and negotiating plausible revisions.

In the kind of knowledge work reported above, there are discernible stages in the process that reflect the progressive nature of the investigations that went on. Because knowledge work of this kind involves changes of mode within each stage, this needs to be reflected in the objects and structures of the computer support system.

**2.1.4.2 An Example from Design**

Individual case histories provide the source material for an analysis of the elements of creative work. The design of LotusSport Bike was chosen because it was a high profile example of innovative design that made the headlines at the 1992 Olympic games. By investigating what lay behind the innovation in terms of the designer's working process, an understanding of the creative processes as a whole was sought. From such a study, one
outcome is a greater understanding of how we might learn to be creative or encourage its development more widely.

Conventional wisdom has it that truly novel ideas spring up suddenly, often without precedent. The LotusSport bicycle design illustrated that the pathway to innovation can be a long and, difficult one. The designer's experimentation and innovation spanned a period of ten years during which time, the bicycle frame was transformed from the conventional diamond shape into a single unit made of carbon fibre material (monocoque). The advent of the monocoque has given rise to a reorientation of the whole concept of what a bicycle frame might be. At the time it first appeared, the design contravened conventional expectations of what a bicycle should look like.

A brief summary of the study follows. The main focus is upon creative cognition: this refers to the combined effect of a number of elements of the designer's process that were critical in the generation of new ideas and in the making of artefacts. In the case reported here, all stages of the design and building were carried out by the designer, Michael Burrows. From the initial generation of ideas, the rough sketching, tube brazing and component-making in steel and aluminium to the testing in competition, he was the initiator and executor throughout. Later, when the carbon fibre frame-making was carried out he engaged the help of others for the mould construction and materials processing.

The design process was very dependent upon personal ways of working and, in particular, "designing in the head" combined with "designing in the hands" was an essential element. The ability to formulate the problem at a high level and devise both systematic and opportunistic strategies for moving forward, were key features because these enabled the designer to break out of the existing conventions of bicycle design. Designing in parallel areas was also very significant in promoting the creative insights that occurred. The innovative design outcome came about as a result of years of deep involvement and immersion in the expert knowledge of the field.
Burrows did not use any form of computer support for his design and engineering. He was able to progress his work within the boundaries of the resources he possessed and achieve a standard that satisfied his personal and business requirements. To go beyond that to high volume production or to turn-round his designs and customise them faster would have required significant changes in resources and, indeed, in his whole style of working. He was aware, nevertheless, of the potential of computer support and did not have particular inhibitions about using new tools and methods except in so far as they were not suitable for the task.

The study suggested that existing CAD systems were unlikely to have been helpful in the particular context, not least because of the highly individualised strategies employed by the designer. In addition to his existing knowledge and expertise, he constantly kept abreast of new events in the field. This was characterised by keeping in close touch with other key players and developing a network of contacts.

The key point is that the use of in-depth domain knowledge and access to state-of-the-art expertise was critical to the designer's creative thinking. Such attributes would not have been supported by existing computer systems. In a representation of creativity "in the large", the elements of the creative design process are interlocking and interdependent activities over time. The creative cognition represented in Figure 2 is comprised of recurring patterns that represent a typical approach to creative work [49].
In the elements of creative cognitive style, facets such as problem formulation, ideas generation, strategies and methods and the application of expert knowledge are included. When combined in a particular way by a particular individual, the result is creative.

- **Ideas Generation**: the starting points were the existing traditions and models but it was making analogies that enabled the extension and transformation of the design space.
- **Problem formulation**: the designer chooses problems with minimal constraints and identifies previously unanswered questions. Taking a higher level view of the problem space is critical to being able to step outside the existing constraints and dare to be radically different.

- **Strategies**: a creative strategy may involve risk taking that can lead to significant gains. Taking opportunistic steps as a strategy for change requires the designer to be receptive to new ideas and be able to devise new methods quickly.

- **Methods**: new methods were acquired only when the need arose. The methods were developed or acquired as a result of the change of strategy.

- **Expert Knowledge**: significant domain knowledge was continually being updated by using a network of contacts with the experts in the field. Breaking information from other innovators rather than documented material is key.

### 2.1.5 Discussion: Requirements for Creativity Support

In this section, issues for computer support to creativity are considered in the light of the results of the case studies described in above. The case studies described above illustrate different approaches to the gathering of evidence about creative work. In the first example, the interaction with domain knowledge is the focal point at a detailed task level. From the second example, a model of creative cognition was derived that encompasses the recurring patterns of the designer's activities. In both cases, the individual's expert knowledge of the field played a significant role in the process. The results of these studies were represented as models of creativity: one of creative process, the other of creative cognition. These models inform the requirements for a Creativity Support System, in which creative work involving a combination of knowledge evaluation and extension, visualisation and collaboration are centrally important.
From the investigations of creative work and the models of cognition and process that have been derived, it is possible to envisage a future creativity support environment that addresses the needs of the creative user. In terms of the functions required, there are three main activities that are fundamental to the user’s thinking and working needs: support for i) Knowledge Evaluation and Extension ii) Visualisation, and iii) Collaboration.

2.1.5.1 Knowledge Evaluation and Extension

Knowledge Evaluation and Extension is the application of domain knowledge in the generation of innovative outcomes, whether they be scientific results, design artefacts or devices. The nature of knowledge intensive tasks that underpin creative work is an important research issue. Recent understanding about this subject has arisen from studies of people carrying out their tasks in real situations rather than in laboratory conditions. It is a subject that has been investigated in a small number of empirical studies [35,38].

The concept of knowledge as a dynamic phenomenon is differentiated from knowledge that has been gathered and stored as information. Something that is characterised as a "body of knowledge" which is not necessarily useful or to a purpose, is a very different notion to that of knowledge that is meaningful and directed towards a purpose. Perkins advocates the notion of "knowledge as design" for teaching and learning to promote critical and creative thinking [49]. This approach emphasises knowledge as something constructed by human enquiry and, by implication continually evolving and, therefore, is particularly relevant to studies of creativity. In the studies reported above, expert knowledge was drawn upon, evaluated and reassessed: but that process was 'creative' in that it often resulted in new knowledge being generated.

The evaluation and extension of domain knowledge is the process of acquiring and evaluating different types of knowledge and relating it to new concepts under consideration. The application of existing knowledge to new concepts involves extension of that knowledge. The knowledge might be both informal (notes and images) and formal (rules and strategies for use). Direct access to all forms of knowledge in the system and a facility to change it in the light of evaluation and experience is an important requirement.
The development of expressive knowledge interaction techniques is a key research area. Providing the user with a repertoire of techniques for interaction with domain knowledge in a way that accords with his or her cognitive style is a need that has yet to be addressed. In addition, the integration of knowledge sources as diverse as geometry, ergonomics, safety, the law and marketing is a significant aspect of research and is especially applicable in concurrent engineering design.

Interaction techniques that enable designers to modify existing knowledge and express new knowledge to the system are needed. Techniques based upon appropriate domain models enable direct interaction between designers and the knowledge represented in the system. This implies providing end user access to the formal structures and rules of the design knowledge in the system using end user programming techniques based upon domain specific formalisms [1]. Being able to interact with the knowledge enables the user to use it for evaluation of alternatives under consideration.

2.1.5.2 Visualisation

Visualisation involves working with visual data such as images, drawings, sketches, diagrams, charts, graphs, graphical objects, that are specific to the domain. It takes the form of expressing ideas and concepts through sketching, annotation and examining multiple or alternative views of the same data, all of which varies according to the domain of interest.

In different fields of work, the use of visual representations, whether it take the form of tables, graphs, pictorial images or drawings is commonplace. The appearance of the object being designed is only one of the many concerns of these representations. Indeed, perspective drawings, for example, are often only used for communication with the client and not as an aid to design itself. However, visual and analytical design activities are to be found in all design domains from product design to engineering design, although the degree of emphasis varies considerably [50].
Whilst, visualisation takes many forms depending upon the domain, its role in design is critical. Current CAD systems provide some forms of visual representation (e.g. 3D rendered models) and alternative representations (e.g. the numerical co-ordinates). However, there are limited methods available for interacting with the full range of required representations. Tools and techniques such as pen-base input devices, expressive sketching techniques, graphical programming, electronic white boards have been developed see [51-53].

2.1.5.3 Collaboration

Collaboration design of complex product development, whether of a new automobile or a software system, are carried out by teams both large and small, often working in parallel. Studies of individual creative people carrying out small scale tasks can only yield limited information. The nature of collaborative work should be investigated because we need to understand more about the way collective knowledge is communicated if we are to provide appropriate support for team work. Such communication, combined with support for information access via the internet, will be an important resource for creative users. However, these developments raise a number of research issues for interactive system designers: for example, how to address the sharing of criteria, the resolution of conflicts and the representation of collaborative knowledge between different designers and teams.

For effective support to human to human communication, there is a need for computer environments that incorporate techniques for the mediation of user communication and input to design activity. Jones and Edmonds [54] describe an environment which supports collaboration between colleagues who are working at different locations. The environment enables designers to participate in a shared environment and work independently at the same time. This enables designers to have an awareness of the design process of the shared artefact and to explore a particular issue privately.
2.1.6 Integrated Creativity Support Environments

The requirements for computer support referred to above have already been realised as discrete elements in a number of applications. It is, however, the potential for significant changes in the total design of a creativity support environment that is the concern here. Some issues to be addressed are:

- All three functions need to exist in combination, as a seamless environment within which the user has complete freedom of movement and data is exchangeable in common formats.
- The interaction methods should be appropriate for the cognitive style and working practices of creative users.
- Existing forms of computer system support remain to be made accessible to those who most need them without the time constraints of office hours and the physical limitations of desk top workstations in respect of portability.

2.1.6.1 Prototypical Design Knowledge Support Environment

An example of a system that employs domain knowledge for concept design support has been developed in order to demonstrate the kind of support environment that might lead the way towards a creativity support environment. The LUTCHI Vehicle Packager Knowledge Support System (VPKSS) [55] has been designed to aid designers at the conceptual stage of the automotive design process. Graphical interactive techniques enable the designer to interact with the domain knowledge in specific concept vehicle design terms. An important aim is to provide that knowledge support to the designer as the design activity proceeds. The knowledge is used to enable the exploration of possible design solutions.

During the course of the designing activity using the VPKSS, the designer may encounter an unintended implication of the modifications to the existing design made. This issue has to be resolved and, in doing so, there is a need to re-formulate the design problem in hand. In order to support this process, the VPKSS enables the designer to analyse the design in progress against a given set of constraints.
The knowledge in the system embodies sets of prototypical designs as well as rules about legal requirements, costs, manufacturing processes and other constraints. Interaction with such knowledge has two effects: first, it provides information about constraints and underlying assumptions (of the system) that are immediately accessible to the designer during the process of developing the design. They can either be observed or ignored by the designer as the design proceeds. Second, it allows the designer to make explicit changes to the formal knowledge in the system i.e. to alter the basic status of the design constraints. This might take the form of changing the ranges of the parameter values or, indeed, adding new parameters or relating existing ones in different ways.

The VPKSS provides access to knowledge as a set of constraints in the form of domain specific rules that can be used to test and generate design ideas as they are being developed. It provides facilities for the consultation of rules in the knowledge base that allows the designer to consider all the elements of the design under consideration and whether these meet existing rules. Where they do not, the designer may alter the rules by adding new constraints or modifying existing rules. It provides facilities for capturing knowledge dynamically as the design proceeds.

2.1.7 Future Research

Future research directions concerning interaction with creativity supporting systems are now considered. The challenge will be to create support environments that go well beyond current concerns for better interaction techniques and the emulation of human cognitive processes.

User Centred Systems Design arose in the 1980s in response to the new demands of personal computing and the manifest failure of many systems to meet the needs of an expanding population of users without technical expertise. The primary focus for the approach was how to take the results of investigations into people, their tasks and work practices into the design practice of system developers. For a time, the user interface
became the main item of attention, both as the solution to achieving user satisfaction and a single unified view of the entire system, as seen from the user's perspective.

In the 1990s, the scope of human computer interaction has embraced collaboration and remote communication across an ever expanding range of specialist domains from science and finance to product and engineering design. In the bridging world between the fields of Human-Computer Interaction and Artificial Intelligence, the notion of the human complementary system, alongside that of the human emulation system is now on the interactive systems research agenda. Thus, research into computer automation and computer support may be reassessed as two different, but not entirely irreconcilable, paths towards understanding the needs of users and ways of generating better designs.

Terveen [56] identifies themes where convergence from existing divergent streams may be possible. The areas for research suggested are:

- **natural communication**: natural language and direct manipulation
- **collaborative adaptation**: adaptive systems and adaptable systems
- **reification**: visible and concrete entities for shared access by user and system
- **balance between representation and reasoning and interaction**.

Some additional requirements for further research are also identified:

- Techniques for integration of different AI techniques with each other and conventional systems to support the user's tasks with various tools.
- Allocation between user and system of automated and mediated tasks.
- Integration and control of co-existing but different views, representation forms, levels of abstraction and complexity.
- The dynamic character of design in terms of strategies, procedures and knowledge application [57].

### 2.1.8 Conclusions

The paper has presented ways in which empirical research in the form of case studies provides us with evidence about creative processes that can inform computer support
system design. A better understanding of creativity comes from investigating people carrying out their work in the situational context of that work rather than as discrete tasks in experimental conditions. Existing computer systems, whilst having many attributes that may be used to support users, have significant limitations when it comes to creativity. Requirements for creativity support systems were discussed and a future research agenda were proposed.

Acknowledgements

The author wishes to thank the following for their support and co-operation to the research described in this paper: Michael Burrows, Ernest Edmonds, Bryan Murray, Stella O'Brien, David Patrick, André Schappo and Lotus Engineering plc.

References


2.2 CREATIVITY IN KNOWLEDGE WORK: REQUIREMENTS FOR SUPPORT


This paper is concerned with the requirements of computer support for creativity. Our objective was to understand the creative process in knowledge intensive work and to draw from that the constraints and possibilities for helpful human-computer interaction. A study of a scientist using a knowledge support system is described. We present the process model of support for creative knowledge work and show how it can be used to specify HCI requirements for computer support.

Keywords: Creativity, Science, Process Model, Computer Support, Knowledge Work

2.2.1 Introduction

There are many ways of characterising the nature of creativity. In this paper, we consider creativity in the context of scientific work, first, as a human creative process, where the individual's knowledge develops as new insights emerge during investigative activities and second, as a creative product, which can take the form of new knowledge or published results of the investigations.

Human tasks can range from the routine and well defined to the exploratory and unpredictable. In the main, the procedures involved in routine tasks can be predicted whereas, creative activities often lead people along new and less predictable paths. By the very nature of creative work, we cannot describe everything that takes place in advance and this makes the modelling task somewhat challenging. In particular, the application of knowledge that is highly expert, distinctive in character and constantly evolving is a feature of the way creative people work. It is the knowledge intensive aspects of creative work that has been the focus of our recent studies.
The aim of the research has been to identify the requirements of computer systems that might support creative knowledge work. The starting point was to try to understand something about the human creative process and to draw from that guidance about the possibilities for helpful human-computer interaction. Our approach was to consider specific instances of possible creative activities in relation to interaction with a particular form of computer support. From that we moved towards an understanding of what the requirements for support systems might be.

In this paper we report upon a study of a scientist using a system that supports the knowledge intensive aspects of scientific work. The example we draw upon here is one where a phonetician used a knowledge based system for the capture and extension of Speech Science. A scientist is an example of the class of people who are "knowledge workers", that is to say the prime concern of their work is to manipulate or generate knowledge. In particular, the knowledge worker not only transforms existing knowledge for others with whom they work or associate but are themselves changed by that process [9]. It is this quality of knowledge work that is relevant to providing support for creativity.

Computer-based support systems for knowledge workers are known as Knowledge Support Systems (KSS) [6]. One of the key characteristics of a KSS is that it allows the end user, in this case the scientist, direct access to the domain knowledge in the system without using the underlying programming language. This is vital because using and extending knowledge is the central concern of such a user. We present a process model of support for creative knowledge work that has arisen from the findings and then show how it can be used to specify requirements for computer support at each stage in the process.

2.2.2 Creativity in Knowledge Work

There has been a growing interest in creativity in recent years. One challenge is to find a scientifically describable explanation of creativity. Boden proposes two categories of creative ideas, concepts, artefacts and styles of thinking: those of historical creativity (H-creative) and psychological creativity (P-creative) [1]. In the former case, ideas are novel with respect
to the whole of human history, i.e. ideas that are first credited with originality. Psychological creativity, on the other hand, occurs within the individual mind, i.e. the person experiences the idea as fundamentally new whether or not others have had the same idea. That person may not be aware of other similar ideas or, indeed, may not recognise the social significance of the idea at the time of its creation.

In Science, creativity is usually acknowledged to be of the H-creative type. If ideas or concepts are recognised as being original contributions to the field, they have, in effect, been accepted into the canon of knowledge by the scientific community. On the other hand, the creative process whereby the scientist achieves new insights, is framed or constrained, by the scientific methods and validation techniques used. There are, however, many examples of a broader and richer pattern of human thinking and action in the generation of scientific knowledge [1,14]. There are many ways of accrediting scientific knowledge. It may be perceived as the product of the scientific community itself which validates claims of new knowledge [10]. What is valued as original knowledge may be dependent for its discovery on recognition by the dominant peer group [11]. By contrast, Simonton, argues that all forms of creativity, especially those in scientific domains, share a common basis for discovery and invention [15].

Scientific knowledge work provides an opportunity for studying the creative process. In order to understand creativity in the context of actual scientific research, there is a need for longitudinal studies of scientists carrying out their investigations in a realistic context. In this way, we can gain more insight into the ways ideas are generated, considered and pursued. In modern science, the use of sophisticated technological tools is becoming standard practice. It is important to include them in any study of scientific process.

Our approach has been to examine scientific knowledge work of the P-Creative type whilst acknowledging that H-Creative criteria are a means of evaluating the outcomes. For that purpose, the use of a KSS, where the knowledge is being captured interactively by the scientist, facilitates the research process in both ways. This is because the knowledge is being made explicit in the system and therefore subject to further scrutiny both by the
scientist who compiles it and any other investigator. If creative work is taking place, the knowledge will be constantly emerging in the mind of the scientist as the experiments progress and results are assessed. Working with incomplete knowledge, by its very nature, involves creative thinking and continuous reflection.

2.2.2.1 Interaction with Computers in Scientific Research

The process of interaction between an expert and a knowledge system can be more than simply capturing knowledge in machine usable form. Creative insights, some of which can be represented in the system, may arise in the mind of the human investigator during the interaction. That interactive process, when part of a broad spectrum of scientific investigations, can provide a mechanism for evaluating, refining and devising ideas. In such tasks, the role of interactive systems incorporating visualisation and knowledge based techniques deserves particular attention.

Scientific knowledge workers can be provided with powerful methods for studying and evaluating their source material using computer support. Graphical techniques for marking up visual phenomena and expressing knowledge about that data in rule form are available. The need for and use of visualisation of data in the scientific domain is now well established.

Visualisation is concerned with the viewing and manipulation of source data. With the advent of advanced 3D graphical computing environments [5], the scientist or engineer or designer may view and manipulate high quality domain specific data such as aircraft models, molecular structures, brain scans or architectural models. The facilities for changing the parameters, and hence the "view", enables the such knowledge workers to explore the visual data in order to understand and interpret it more effectively.

In the context of the research reported here, where the visual data is the continuous speech utterance (represented as a spectrogram wide band signal), visualisation is widely recognised to be an important means of expressing humanly perceived knowledge to a computer system [4]. In the KSS used in the study described below, the visualisation of such data is combined with explicit rules about the associated knowledge that has been compiled by the scientist.
The scientist expresses his or her interpretation of the visual data using graphical methods and this then becomes a formal description of expert knowledge about the visual data that is incorporated into the knowledge system.

To enable the scientist to express knowledge to the system in a domain specific language, a bridge between the representation of visual data and the representation of scientific theory in the form of rules was made available. In effect, the visual data analysis and interpretation is combined with methods for capturing formal domain knowledge as rules. The scientist uses graphical interaction techniques to annotate visual data and to specify the knowledge gained from studying the data. That specification is then automatically incorporated in a knowledge-base without the user having to use the underlying code. Having assimilated the results from the exploration of the visual data and the knowledge base, the scientist can continue to advance existing knowledge further as the analysis and interpretation progresses. The aim is to reduce the constraints upon the scientist's explorations and unpredictable courses of action.

2.2.3 Computer Support for Science: A Study

In this section we describe the results of a study of a scientist's practice and the role of a Knowledge Support System in that process. The aim was to identify the key features of human computer interaction in the use of a KSS for scientific knowledge work. The characteristics of the observed process are used to hypothesise about requirements for computer support to creative knowledge work.

The goal was to gather as rich a set of data as possible within the constraints of the normal working situation: i.e. this was a study of on-going scientific activities rather than a selection of experimental tasks. The study methods employed were those of direct observation, monitoring and interviews over a continuous period of six months. The whole data set included observations of the scientific activities and interactive process including video and audio recordings of the detailed work. It also included the scientist's own reflections upon her
scientific goals and methods and the experience of interacting with the KSS. A full account of the study methods, data analysis and results is described by Candy et al [2].

The scientist's main goal was to study the contribution of a set of visual features to the identification of continuous speech data. The scientist's method was to analyse visual source images (spectrograms) and then test the findings by an identification from a knowledge base. The annotation of the visual data allowed the scientist to express what were perceived to be the basic facts about the features of the speech images. These were expressed in machine-readable form. In effect, they represented a set of hypotheses about the data which were analysed using the existing knowledge base. This analysis generated an identification of the speech utterance in the form of a phoneme lattice.

In this way, the scientist captured knowledge in a rigorous form and, in the process of externalising it and having the machine process it, identified implicitly held knowledge. The scientist identified unexpected features of the visual data and revised her current expectations (theories) about the interpretation of that data. The results are described by O'Brien [12,13].

The large volume of speech data that could be examined and evaluated in a very short space of time using the KSS had an influence on the scientist's research strategy. Possible new avenues of investigation were devised and then pursued. This began with the designing and recording of an entirely new set of visual speech data with more complex features. In turn, this strategy gave rise to insights into the knowledge being applied to the data, and the methods and tools being used to carry this out, including the design of the graphical techniques being used.

The process of designing, analysing and testing data was, in the first instance, primarily a linear one. However, as the investigations proceeded and the testing of the knowledge captured in the system took place, it became iterative. When inconsistencies and unexpected results arose, the scientist moved between image analysis, testing and refining the rules, in parallel. The key activities in the whole process are described briefly below. In general the
overall order of the three main stages of the process followed a stable pattern within which changes in the scientist's modes of thinking and action took place.

### 2.2.3.1 Visual Data and Testing Existing Knowledge

The scientist's first task was to identify the features of continuous speech recognisable in the visual image of the speech utterance. Initial knowledge about the relationship between visual features, and their contextual relationships, and speech events had already been expressed as rules in a knowledge base. A set of such rules that applied to certain speech phenomena was then selected and, by asking for an analysis of the data, tested for accuracy. The identification successes or failures were then evaluated by the scientist.

The results of the analysis by the knowledge base appeared as a set of all possible hypotheses for each segment of the speech utterance image. The scientist considered the whole utterance first to see if there were discernible patterns across the complete image that warranted particular attention. Following that, each segment was assessed one by one and the rules applied to each identification examined by the scientist. Movement back and forth between the whole image and the close ups of individual segments took place continually. This stage gave rise to refinement in rules with further testing: the cycle of test, refine, modify and test was applied repeatedly.

### 2.2.3.2 Multiple Views in Visual Data Interpretation

The analysis of the speech image was a cyclical process which involved very close scrutiny of that visual data. In the main, it was a routine procedure where the intention was to base all judgements on visual criteria only. A rapid segmentation in the time dimension across the whole image was followed by the application of criteria to obvious features which were labelled with appropriate graphical objects. Once completed, this paved the way for further attention to the uncertainties and ambiguities identified previously but reserved for later assessment. Where unpredictable features occurred, this prevented a routine response and the typical pattern of graphical annotation was disrupted. The scientist's judgements were affected by this and also by the quality of the visual data. Without clarity of visual
information, speculation as to the likely character of the features arose, with frequent reference to high level knowledge (e.g. knowing that some patterns occur always even though they are not visible on the image). Alternative views of the visual data that gave support to "intuitive" opinion (e.g. low frequency energy distribution graphs and cross-sectional profiles) were used frequently to arrive at a judgement more quickly.

In summary, the KSS supported rapid graphical interaction with the visual images. This was then recorded for later use at the testing stage when the user asked for an identification of the annotated speech utterance by the knowledge base. The marking of the visual features using colour, blocks and lines appeared to heighten the scientist's awareness of the finer details of the image. The overlaying of alternative quantitative information about the visual data supported more reliable judgements. The speed, quality and quantity of visual data access was vital as multiple images were required simultaneously and the interchange between them needed to be quick and fluent. Unpredictable features were identified that could not be marked up using the standard techniques and the scientist used an improvised solution. This often led to a need to refine existing rules.

### 2.2.3.3 Revising and Creating New Rules

Having evaluated the results of the knowledge base identification, the scientist could then change the rules or add new ones according to the fresh insights which arose during this process. For example, having identified the need to take account of the effects of contextual variants on the recognition of particular visual features, new rules were created, tested and evaluated about those effects. In total, the process was an iterative one in which ideas were applied, abandoned or refined as the discovery of new insights took place. When a solution, in the form of a refined rule, proved impossible to achieve, the existing rule or rules was typically discarded. It was then that the whole problem was reformulated and an entirely new rule created.

In a series of experiments, the eventual outcome was that the scientist advanced her own understanding of the speech science knowledge she was using. A number of issues were also
identified that required further investigation. For example, the need to be able to structure 
rules in groups and hierarchies was recognised but how to achieve this depended on a much 
clearer understanding of how the domain knowledge should be applied.

2.2.4 Observations on the Role of the Knowledge Support System

The KSS contained within it a set of constraints within which the scientist set out to achieve 
certain goals. The extent to which the scientist moved beyond the boundaries of those 
constraints and created new concepts can be seen as an indication of the creativity involved. 
Whilst we have concentrated on the P-Creative aspects, H-creative results arose that were 
documented in published material [12].

The methods of interaction were designed to enable the knowledge worker to manipulate the 
visual source data and the knowledge base in a flexible manner. Many questions that were 
unforeseen in the early plans arose. The fact that the scientist was able to use the system to 
extend the scope of her experimental studies beyond the original intentions enabled her to 
evolve her understanding of the knowledge being applied.

The fact that the feedback on the knowledge being applied was rapid and immediate was 
useful but that was not the main advantage. The rules were created and then evaluated against 
the source data under scrutiny by requesting an identification. This implied that the speech 
knowledge captured was tested rigorously. Because it represented the current state of the 
scientist's thinking, the continual process of confirmation or refutation as the results of the 
identification process were produced was quite powerful. This was a form of support that 
challenged the scientist's hypotheses and forced a response at the knowledge level not 
merely the surface interaction. A more detailed discussion of the above results is to be found 
in [2].

The findings of the study gave rise to a new framework for computer support to creative 
knowledge work. In the kind of knowledge work reported above, there are discernible stages 
in the process that reflect the progressive nature of the investigations that went on. Because
knowledge work of this kind involves changes of mode within each stage, this needs to be reflected in the support system. For that purpose, we need to have a model of the overall process and, within each stage, a set of requirements for the system design that supports each mode of activity. The process model is described in the next section followed by proposed requirements for supporting the different modes of human activity.

2.2.5 A Process Model

The creative scientific process observed in the above study may be described in terms of three main stages which the knowledge worker moves between: generation and invention, exploration and evaluation and the consideration of constraints and requirements (Figure 1). Our particular focus is on the interactions between the knowledge worker's actions and those of the KSS as derived from the study.

![Figure 1: A Process Model](image)

Taking each of those in turn, we can decompose them and consider the roles of knowledge worker and KSS more specifically. The exploration and evaluation activity consists of examining the data, evaluating and refining the rules (Figure 2). In this case, the balance between human and computer system shifts during the process. Whilst the human examines the data and refines the rules that the system presents, the data is analysed by the system according to the existing rules which are, as a consequence, evaluated by the human.
Examine data

Analyse & Evaluate

Exploreation & Evaluation

Refine Rules

Figure 2 Exploration and Evaluation

Generation & Invention

Examine Data

Create Rule

Insight

Figure 3 Generation and Invention
Generation and invention involves, in addition, moments of insight and the creation of new rules (Figure 3). Both of these are human activities. In particular, any insight obtained is an insight of the human understanding.

The consideration of constraints and requirements, on the other hand, involves receiving and clarifying them, revising, and possibly negotiating, the revisions (Figure 4). Here, then, the system presents the existing constraints and requirements, the human revises them and the resources of both are employed in the process of considering and negotiating plausible revisions.

![Figure 4 Interpretation and Reformulation](image)

We can now place the findings from our study in relation to the Process Model [2,6]. The requirements proposed in the section below are drawn from a scientific knowledge work example. We can hypothesise that they may also be applicable to creative knowledge work in general. With that proviso, we present a structured view of proposed requirements for a knowledge support system.

### 2.2.6 HCI Requirements and the Model

Criteria for the requirements to support creativity have been proposed by Fischer [7] in the context of the Design domain. Fox [8] identifies desirable attributes for decision support such as the ability to reflect upon knowledge and the investigation process, to reason about the inferencing methods used and the need for explanations to be couched in domain familiar descriptive objects. We have drawn upon such comparative work and extended the ideas in the light of our studies and experience.
What follows is proposed as a structured checklist for HCI designers of systems intended to support knowledge workers. Not all requirements will apply in every case but for the system designer each one implies a question to be addressed. The modes within each stage of the process are considered in turn. Some requirements apply to more than one mode and are, therefore, repeated. Where this occurs only the headings appear.

2.2.6.1 **Exploration and Evaluation**

**Examine**

- **Holistic Views**

  Holistic views of high quality visual data that can be manipulated and annotated should be available to the user.

- **Multiple Representations of Data**

  Access to multiple representations of the data which can be used to support the user's judgements should be provided.

- **Visual Data Annotation**

  Annotation of high quality visual data which can be incorporated into the knowledge activity should be provided.

- **Concurrent Processes**

  Access to the different forms of visual data and the methods for knowledge base interaction should support concurrent use and enable the user to switch between activities fluently and quickly.

**Evaluate**

- **Multiple Representations of Knowledge**

  A plurality of representations should be available so that new knowledge structures that emerge as a result of changes in the user's understanding can be readily incorporated into his or her activities.

- **Feedback**

  Evaluation of knowledge by rapid feedback should be supported.

- **Domain Specific Evaluation**

  Domain specific support to evaluation with explanation about negative and positive results should be available.
Refine

- 'Natural' Graphical Interaction

Interaction with knowledge in the system should use graphical techniques and draw upon images and terminology natural to the user and the domain orientation.

- Knowledge Modification and Evolution

Accessible and powerful methods for modifying the knowledge base should be provided so that the user can refine existing rules and add new rules incrementally at any time during the on-going tasks.

2.2.6.2 Generation and Invention

Examine

- Holistic View

- Multiple Representations of Data

- Concurrent Processes

- Evaluation of Evolving Knowledge

Support for evaluation of the evolving knowledge in progress should be provided. The user must be able to ask why or why not about the results of any request for evaluation and receive an explanation that is expressed in domain specific terms.

Create

- Creating Objects

Facilities for creating graphical objects or icons in addition to pre-specified objects that can be used to express knowledge about new visual features should be provided.

- Knowledge Modification and Evaluation

Accessible and powerful methods for modifying the knowledge base should be provided so that the user can define, modify and add new rules incrementally at any time during the on-going tasks.
Comparative Evaluation of Knowledge

Support for comparative evaluation of the knowledge in the system should be available. The user should be able to create and evaluate selected rule sets with different variables against identical experimental criteria allowing immediate comparisons of their performance relative to each other and to a standard rule base.

2.2.6.3 Constraints

Receive and Revise

- 'Natural' Graphical Interaction
- Knowledge Modification and Evaluation
- Comparative Evaluation of Knowledge

Negotiate

- Knowledge Modification and Evolution
- Visual Data Annotation
- Comparative Evaluation of Knowledge

In summary, for the interaction between user and knowledge system to be effective at the more creative end of the spectrum, it must support the extension of knowledge and be in accordance with the user's domain orientation. Depending on the context, there will be variations on the basic set of requirements. However, some features are likely to be vital: for example, working directly from visual source data (e.g. X-rays, scene drawings), having suitable graphical methods for marking the visual data and being able to describe domain specific ideas in an accessible and appropriate way. Other support to knowledge analysis and evaluation, such as being able to access statistical packages and carry out tests, may also be necessary. In particular, the interaction must not be constrained by the need to use the notation of a particular programming language whilst, nevertheless, providing access to the full power of the system's functions and facilities especially in respect of the domain knowledge.
2.2.7 Conclusions

The objectives of the work reported were to identify requirements for the design of creativity supporting systems. It is clear that such support systems must be designed to provide the maximum flexibility for the user to handle and extend knowledge. A need to make available different forms of representation of the knowledge was identified. A plurality of approaches is essential if the scientist is to be able to consider and control a number of knowledge sources. The findings of the study have given rise to a process model which enables us to delineate the role and function of Knowledge Support Systems at each stage in the process and to identify the specific requirements. The findings are not proven to be generic but sufficient evidence exists for us to hypothesise that they have a broad applicability to the design of systems that support creative, knowledge intensive tasks. The ideal system is a fully fledged workbench that provides a repertoire of tools and resources to support the emergence of new knowledge. Studies have been carried out in Design and a KSS applied in that domain [3].

Acknowledgements

The work was partly funded by the UK Cognitive Science/ HCI Research Initiative. We are grateful in particular to Trevor Foster, Eddie McDaid and Stella O'Brien whose work was so vital to the project's success.

References


2.3 INTERACTIVE KNOWLEDGE SUPPORT TO CONCEPTUAL DESIGN


The paper addresses the question of how to support the designer with appropriate knowledge during conceptual design. It begins with a discussion of knowledge-based support for product design and is followed by a scenario account of the use of a knowledge support system. The approach has been demonstrated in an architecture and a system that applies to the task of vehicle packaging. The LUTCHI Vehicle Packager Knowledge Support System (VPKSS) has been designed to demonstrate methods that aid designers at the conceptual stage of the automotive design process. It supports the creation of new designs by way of a solution generation and evaluation process that relies upon cooperation between the designer and the knowledge system. Graphical interactive techniques enable the designer to interact with the domain knowledge in specific concept vehicle design terms. Evaluation and future directions are then discussed, in particular, the extension of the system support to a multi-user platform to support team working.

Keywords: conceptual design, knowledge-based engineering, computer support, knowledge support systems

2.3.1 Introduction

In the work described in this paper, we are addressing the question of how to support the designer with appropriate knowledge during conceptual design. An important aim is to provide that knowledge support to the designer as the design activity proceeds. The knowledge is used to enable the exploration of possible design solutions by the provision of generative and evaluative capabilities in the system. These requirements have been demonstrated in an architecture and a support system that applies to the task of vehicle design.
We have developed an approach to design support that uses a new kind of environment called an interactive Knowledge Support System. Support for interactive knowledge-based design requires fluent interaction between designer and knowledge in a way that does not impede the conceptual design process. From a computer support perspective, this implies a number of needs such as, multiple and parallel viewing of information and flexible interactive facilities to handle the information.

The key activities include:

- knowledge acquisition from the designer during the design process
- knowledge maintenance that is a continuous process of refinement and updating
- graphical interaction techniques that draw upon domain specific terms.

The LUTCHI Vehicle Packager Knowledge Support System (VPKSS) is a demonstrator system for aiding designers at the conceptual stage of the vehicle design process. It supports the creation of new designs by way of a solution generation and evaluation process that relies upon co-operation between the designer and the knowledge system.

The paper begins with a discussion of support for the conceptual phase of the design process and knowledge-based approaches to design. We then propose a set of requirements for knowledge support to conceptual design that have been identified from previous work. This is followed by a scenario account of the use of a knowledge support system for vehicle concept design. The graphical features of the system and its architecture are then described. Finally, future research developments are proposed, in particular, the extension of the system to a multi-user platform to support team working.

2.3.2 Conceptual Design and Knowledge-Based Approaches

There have been many attempts to define conceptual design and the variation in perspectives often depends upon very fundamental issues, such as whether it is based upon a scientific rationalist tradition or it invokes non-rationalist views [1]. In practice, approaches to the whole design process itself are undergoing revision rapidly. This may be
partly a result of changes in the customer requirements or market imperatives and partly a result of new technological support tools and methods. In that context, conceptual design may include the requirements specification as well as the early design solutions. For example, Bowman and Cooper characterise the task of the conceptual designer as, "to understand the customer's need, analyse it and produce a model of the possible solutions and present these for the customer's choice/acceptance". They also distinguish it from detail design thus, "The object, therefore, of concept design is to describe the total product, that of detail design to describe how that product might be made." p12 [2].

In Engineering Design, conceptual design has been characterised as that part of the design process which succeeds the stage when the essential problems are identified and the requirements defined [3, 4]. The designing activity involves the formulation of solutions or concepts that meet those requirements before going on to embodiment and detail design. Here, there is consensus amongst the so-called "prescriptive" design models [5]. However, the exact nature of the negotiation that occurs between the various stages from the initial problem definition and concept design stages through to detail design and the different strategies adopted by designers remains a research issue.

The domain-orientation of design process models in current use has not been the explicit subject of research although it is plain from the models themselves that their character is influenced by such factors. Thus, for example, differences between the engineering and industrial design are evident in respect of the importance placed on visual design activities, which are more prominent in the latter case [6]. The advent of concurrent engineering has longer term implications for the integration of all aspects of design from the visual (and spatial) design processes to numeric aspects. In respect of vehicle design, the vehicle package provides a total product model to which a knowledge-based engineering approach can be applied.

The approach to conceptual design reported in this paper is one set within a framework of Knowledge-Based Engineering (KBE). Claims that KBE systems offer significant improvements in the time taken to engineer complex assemblies have been made and,
although the gains have not been documented in any detail, the approach is gaining ground rapidly [7]. In KBE, an important aim is to capture knowledge about the design, analysis and manufacture of a product so as to represent the engineering intent behind the design in a systematic product model [8].

A primary goal of KBE is to maximise the synergy between the design concept and the manufactured product. In the lengthy process from conceptual design through specification to manufacturing, it is often the case that the original design is not carried through because it turns out not to be feasible at the implementation stage. The changes or compromises made in manufacture, however, may not reflect the designer's original intentions, because their full implications are not (and probably cannot be) transmitted through the process.

Knowledge-based techniques are being applied in order to support concurrent design engineering with the aim of improving overall cost effectiveness in design [7]. These techniques provide an opportunity to integrate all the elements of the design process in one model. The use of rules makes it possible to incorporate many diverse design considerations. e.g. costs, manufacturing capability, legal requirements etc. When all the significant considerations have been expressed in rule form, the designs may be checked and, where not feasible or legal, may be amended at an early stage before additional commitment is made.

In respect of vehicle design, it could be argued that there is no advantage in providing computer support to the concept design styling stage because the time-scales are very short and no significant lead time can be gained. In the design development stage of vehicle design, there are, by contrast, obvious opportunities for adopting a knowledge-based approach [9]. This stage lasts longer and overlaps with computerised manufacturing procedures. However, if the goal is to tighten the loop between the concept design and detail design stages, there is a need to facilitate a total design approach.

The case for paying attention to optimising the quality and accuracy of the concept design rests upon the heavy resource commitments made at this stage that have significant cost
implications for changes downstream. In vehicle design, the basis of a total or concurrent design approach is already in place in the form of the vehicle package where the attention to body styling and structure, (including the passenger compartment) takes place in parallel with engineering considerations. New developments are underway in a number of design applications [7].

One such approach to knowledge-based support for concurrent engineering design is that used in the ICAD system [10]. At Lotus Engineering, a Vehicle Packaging System (VPS) based upon ICAD [11], has been developed which supports the conceptual design of the whole vehicle. In that version of this system, the constraints are agreed first and then the designer is provided with a solution. Such an approach requires the designer to identify and input all parameters and variables beforehand. This is demanding, especially for the less experienced designer. However, the advantage is that the inter-dependencies are well-defined and for that reason, the designs are more reliable. Most significantly, the feedback from constraints to solution is fast enough to enable many iterations to take place.

One of the key aspects of the work described in this paper is the nature of such interactions between the designer and the support system. It is necessary, we believe, to consider these interactions within the context of the design activity that the individual designer is engaged in. It is important therefore, to be clear about the role of the individual designer in the design process as distinct from the organisational procedures within which that activity takes place.

2.3.3 Knowledge Support and the Design Process

We have argued in a previous paper [12] that the variation in design process models can be understood best in terms of the different levels of concern of the models: in other words, whether the intention is to reflect or prescribe the organisational process or the individual designer's process. Often the two are not distinguished where the models are highly abstract. We have represented the individual designer's process as a spiral, in which the emphasis shifts as the design ideas develop and the total range of activities range from
initial idea generation to the production of physical models (see Figure 1 below). In certain respects, the spiral can be seen as a synthesis of other models of the design process. However, for the purposes of the discussion here, the main focus is upon the individual designer process.

![Figure 1 - The spiral design process model](image)

2.3.3.1 Requirements for Knowledge Support

If designers are to be provided with support that will enable them to generate a set of possible designs and to evaluate them before committing to a particular solution, they require knowledge that is provided early on in the process and at the appropriate point during the design activity rather than at the end. For example, the Alias surface modelling system requires that the geometric constraints be checked and modified afterwards rather than during the actual constraint specification activity [13].

In the early stages of design, the designer's initial work is not easily characterised by formal procedures. Hence, computer systems that provide support to formally defined processes are unlikely to be helpful to the conceptual and exploratory stages. Flexible and usable support implies taking account of the cognitive attributes of the designer during the process. A number of such issues in relation to the implied requirements have been identified in previous work [14]. From that and related work, core research requirements for knowledge-based support to conceptual design have been identified as follows:-.
i) Access to Knowledge: Acquisition and Evaluation

A key issue is interactive knowledge acquisition and validation. Interfaces that enable designers to express their knowledge to the system are needed for that purpose. Visual and graphical interaction techniques based upon appropriate domain models enable direct interaction between designers and the knowledge represented in the system. Being able to interact with the knowledge enables the designer to apply it in design and to use it for evaluation.

ii) Development of Knowledge: Problem Formulation

Methods are necessary to support the designer in identifying and formulating the design problems to be solved. Very often, once a problem is fully stated, automatic methods can be employed in its solution. The designer requires support to design exploration that can grow as the design develops. Knowledge used in the act of designing is dynamic and in order to capture it effectively it must be handled during the process, not as an after thought. Indeed knowledge development is an essential part of the process.

iii) Reformulation of Knowledge: the Emergence of Concepts

A key requirement is the opportunity for being able to respond to change. Designers need an opportunity to come back and reformulate the design in a flexible manner. During the process new concepts emerge. For example the addition of a new parameter or the conversion of a constant into a variable may play an important role in innovation. The tracking of the designer's emerging ideas is a significant research problem [15].

The discussion and scenario that follows takes a number of steps towards meeting these core requirements.

2.3.3.2 Knowledge System Support for Conceptual Design

Most conventional CAD models contain geometric modelling information only and do not provide the designer with support for the optimisation of the design: for example, information about design rules and procedures, legal requirements, engineering and quality standards, tooling, costs etc. Because the process of generating a choice of designs and
evaluating them against a set of criteria is not given explicit support, the designer is more likely to seek a best fit solution. This can imply a form of "premature closure" where there is a failure to push beyond initial ideas and find more options. There is some evidence that this might explain why many ideas and products do not reach the level of innovation and originality that the effort would lead one to expect [16]. Where the aim is to provide computer-based support for conceptual design, it implies addressing the more creative and exploratory processes in which the designer is engaged. This suggests a need to provide forms of knowledge that may be used during the actual process of design at the appropriate time [17]. To that end, more expressive interaction techniques that do not impair creativity in design are required.

The nature of the design process is solution-oriented in that the designer analyses general problems by way of the generation of solutions to the immediate problem in hand [18]. There will be a set of requirements and a given set of constraints depending upon the initiating events and the organisational situation. During the course of the designing activity, the designer may encounter an unintended implication of the modifications to the existing design made. This issue has to be resolved and, in doing so, there is a need to re-formulate the design problem in hand. The designer is not simply applying well understood knowledge to a well defined problem, development is taking place in the designer's knowledge. Thus, the evolution of a design is an exploratory activity and there will be unforeseen developments as the designer pursues ideas in detail [19].

In order to support this process, a Knowledge Support System [14, 20] enables the designer to analyse the design in progress against a given set of constraints. The knowledge in the system embodies sets of prototypical designs as well as rules about legal requirements, costs, manufacturing processes and other constraints. The designer thus, has available knowledge about a range of issues that may or may not be useful to the evolution of an individual new design. This may, at any given point, bring to bear considerations in respect of the design that impel the designer towards rethinking the original problem. Indeed, a complete reformulation may then take place and have consequences for the design under consideration.
Where the knowledge is considered by the designer to be inappropriate, he or she may make a conscious decision to ignore some elements of the existing knowledge base. The system reasons on the basis of the new values and provides the designer with a further analysis of the implications of those changes. The designer can continue to change values or to leave them as they are and make the design conform to them. Thus, interaction with the knowledge has two effects:

i) it provides information about constraints and underlying assumptions (of the system) that are immediately accessible to the designer during the process of developing the design. They can either be observed or ignored by the designer as the design proceeds.

ii) it allows the designer to make explicit changes to the formal knowledge in the system i.e. to alter the basic status of the design constraints. This might take the form of changing the ranges of the parameter values or, indeed, adding new parameters or relating existing ones in different ways.

In the approach described below, the requirement to support early design ideas by offering both generative and evaluative capabilities in the system is addressed in the following ways:

i) It provides access to knowledge as a set of constraints in the form of domain specific rules that can be used to test and generate design ideas as they are being developed.

ii) It provides facilities for the consultation of rules in the knowledge base that allows the designer to consider all the elements of the design under consideration and whether these meet existing rules. Where they do not, the designer may alter the rules by adding new constraints or modifying existing rules. In this way the knowledge may be refined and extended.

iii) It provides facilities for capturing knowledge dynamically as the design proceeds. The assumption is that, in conceptual design, the designer's ideas are always developing because the generation of new design solution is a major goal.

iv) These features and their use will be elaborated in the following section in the context of concept design using a vehicle packaging approach.
2.3.4 Knowledge Based Approaches to Concept Vehicle Packaging

Concept vehicle packaging is an approach to whole vehicle design that incorporates a representation of the key elements or sub-systems such as body structure and style, passenger compartment and drive-line. As such, it is an inter-related set of activities involving the manipulation of the major components and multiple sub-systems. The fundamental requirements of the overall vehicle design are described in terms of the operating characteristics of the whole. Creating a specific design solution to provide the required characteristics is, therefore, very complex, and involves compromise at many points. Changes in order to optimise particular parts of the design demand a very large amount of design effort to rearrange all other parts of the design. The scale of this effort tends to restrict the number of optimisation iterations which can be undertaken without effective automation.

2.3.4.1 The Lotus Vehicle Packager System

In the Lotus Vehicle Packager system (LVP) [11] data and rules about vehicle packaging are provided in order that the designer can explore the package by specifying the minimum number of attributes, the system being left to deduce the implementations. The attributes are typically numerical values, such as the required wheel base.

In a typical scenario, the designer approaches the LVP design in a given task order which has been devised specifically for this system. The order is partly determined by the interactions between the vehicle sub-systems and partly by the priorities set by the client organisation. In a typical class of car, for example, the passenger cell might be of primary importance in driving the overall design, whilst, in other cases, the engine and power train might have a higher priority.

Given the required values, which may be defaults, the system generates and displays the vehicle package. Values may then be varied. The process enables and encourages iterations. Because the system generates designs according to the rules, there is no need to check, for
example, that legal requirements are met. Providing that the legal rules are included, only legal solutions will be generated.

The LVP system has demonstrated that a knowledge-based approach can support concurrent engineering design. Whether or not the type system has the makings of a support tool for conceptual design is not yet proven. The ease with which the organisation, and in particular, its design section, can maintain the knowledge, even during the process of investigating a particular design, is an important question. This is because one way in which innovation often occurs is by posing such questions as, "What if we changed that rule?" [21]. Knowledge Support Systems offer a way forward here. In such systems, domain specific interaction languages are used to provide experts with direct access to knowledge within the system and facilities for updating and augmenting the rule base.

2.3.4.2 A Knowledge System Approach

A demonstration Knowledge Support System has been developed at the LUTCHI Research Centre that shows, amongst other things, an approach to knowledge acquisition and maintenance in design. The designer is provided with access to historical knowledge, the rules of received wisdom and databases of technical information. A key element of the approach is to enable the designer to review, evaluate and modify the knowledge in the system. A description of the way the KSS supports approaches to the exploratory and analysis activities in conceptual design follows.

*Exploration and Analysis in Concept Design:* Let us envisage a situation in which a designer wishes to change part of a previous design. This may be to conform to some standard or to evaluate the impact the changes will make on the overall design. The designer displays an existing design represented in the system by an image, possibly even an old sketch. Graphical objects represent individual features of the new design, for example, in vehicle design, the fuel tank, battery, spare wheel, seating positions of the passengers, etc., which can be manipulated by the designer. The system may also make available graphical objects which the designer can use to represent elements of the knowledge or facts within the
system, for example, the rear chair height or the distance between the hips of the front and rear passengers.

At any time, during a design in progress the designer can interrogate the existing knowledge in the system. Here, the designer supplies known parameters and values of the given features and the parameter of the unknown value required. It is assumed that the designer has a good understanding of this kind of knowledge. When the designer confirms the new design decisions, the system then generates a set of parameters and values using the graphical objects which represent elements of the knowledge. The parameter is the identifier of the object and the value may be, for example, the length of the object. The system uses these facts together with a selected knowledge-base to analyse the new design.

For each design parameter the system checks against the existing knowledge base. If no match between the value in the knowledge and the value of the design parameter is found, then it stores the parameter and the value in the knowledge. Once all design parameters and values are analysed the system checks for the completeness of the design. This is done by finding parameters which have been omitted from the new design. Default values may be automatically substituted. In this way, the basis for making the designer aware of the underlying implications of any design change is derived. This supports the drive for a closer relationship between early design and design feasibility.

If, for example, the results of the analysis shows that the value of the parameter has not been matched against the value in the knowledge base, there are two options open to the designer. He or she may change the design to conform to the existing rules, or alternatively, the knowledge may be refined to include the new design attributes. Where the designer chooses to refine the knowledge, he or she may not necessarily know which rule element does not match with the design parameters. A trace facility, which provides a means of interrogating the knowledge in the system to see which rule and elements have not been matched, is provided for that purpose. Once the appropriate knowledge base has been selected, the system provides the designer with a list of all the rules in the knowledge base. The designer selects the rule to be refined and is provided with a rule editor. The trace
shows which element did not match against the design parameters, and also the value of that parameter generated from the design. In order to refine the knowledge to conform with the features of the design, the designer may simply change the value in the rule and if confirm later may be then included in the knowledge-base. In the following section, the Vehicle Packager Knowledge Support System architecture and modules are described. In addition, the approach adopted to graphical interaction is summarised.

2.3.5 The Vehicle Packager Architecture

In this section, we describe an architecture that facilitates the support for conceptual design discussed above. A number of subsidiary facilities, such as the notepad, that are supportive of knowledge development are included in the description for completeness. Each system module, described below, can be used independently or as a whole. For example, the designer may use the analysis module to view the results of a previous design option or he may use the analysis module along with the knowledge base module to view the implications of changing elements in the knowledge. Figure 2 illustrates the architecture of the Vehicle Packager.

Figure 2 - The architecture of the VPKSS
2.3.5.1 System Modules

i) The *Annotation module* provides access to libraries of previous designs, sketches, images and domain specific annotation objects to enable the designer to explore new designs. The designer can select and manipulate these objects to create new designs. The design options are shown in Figure 3 below which illustrates the annotation of a vehicle packaging drawing.

The designer displays an existing design represented by an image in the system. To explore new design decisions the designer places graphical objects over the image. There are two types of graphical objects. The first type of graphical object can be manipulated to represent features of the new design, for example, the seating position of the rear passenger or the position and shape of the fuel tank. The second type of graphical object can not be manipulated by the designer. These objects represent parameters in the knowledge, for example, the rear chair height or the distance between the hips of the front and rear passenger, known as the couple distance. This type of object attach to existing objects, and change dynamically as

![Figure 3 - Annotation of a drawing](image-url)
the design evolves. For example, the couple distance depends upon the front and rear passengers and will change if either passenger hip points are moved.

During the design process the designer may wish to know values for various features of the new design. The system provides a facility which enables the designer to interrogate the knowledge within the system. If, for example, the designer does not know the value for the percentile of the rear dummy he may provide the system with known values for the vertical displacement, the horizontal displacement and the parameter of the unknown value. Once the designer is ready to confirm the design decisions, the system identifies graphical objects which represent the knowledge within the system and generates values from the information supplied by the objects.

ii) The Analysis module uses the specified knowledge base to give feedback to the designer about characteristics of the design. The designer can explore the feedback analysis to identify any mismatches which occurred between the knowledge and the design. The trace facility shows the designer which element of a rule does not match the values from the new design features. The designer may then change the design to conform with the knowledge or refine the knowledge to conform with the new design decisions. The output of a trace identifies items of knowledge not matched during the analysis.

iii) The Knowledge Base module provides access to the knowledge allowing the user to browse, edit and add new rules using graphical editors and indexes. The rule structure is dynamic and the user can change the structure of the knowledge using a graphical specification. Once the appropriate knowledge base is selected, the designer is provided with a list of all the rules contained within it. This is the rule index. The designer can then add, remove or alter rules and review the whole knowledge base using the rule editor. (See Figure 4).
The VPKSS provides facilities to support knowledge maintenance. These include the ability to copy entire rule constructs and rule elements between individual rule editors, therefore, the designer can construct rules quickly with fewer errors. The designer may copy rules and rule elements between rule editors. The elements of a rule which are displayed within a rule editor are graphical text objects. They can be manipulated in the same way as the schematic graphical objects. That is, the designer can select, move or delete rule elements at will.

iv) The Notepad module enables the user to attach notes about ideas, explanations, sources of reference etc., to any individual item of knowledge in the system. For example, images, and knowledge are indexed contextually with reference to the design knowledge and the favoured representation. The designer can attach notes to elements of the whole system. For example, he or she may attach notes to graphical objects, complete designs and rules in the knowledge base. The system provides a note pad editor which enables the designer to create the documents quickly and create links to other note pad documents. Hypertext documents are created from the text and links which have been entered by the designer and the system allows the user to create links to other hypertext documents [22]. Modules can
communicate with the note pad module via the FOCUS messaging system to view the documents at any time (see 2.3.5.2 below).

v) The Feature Designer enables the designer to graphically create new objects or ‘features’ for new designs to be used in the Annotation Module. The system enables the designer to apply constraints to graphical objects that define how they will be drawn and related to existing objects. The designer may add attributes to the primitives of objects and the object itself: for example, line thickness, line style, line colour, text font/size, the objects colour, and how the object will interact with the background. Graphical objects can be defined as being transparent or opaque. Further details about the Feature Designer can be found in Murray et al [23].

2.3.5.2 Abstract Interaction Objects
The Vehicle Packager Knowledge Support System was developed using the FOCUS Front End system development software. It consists of several modules running as separate processes that may be located on different machines, connected via a network. Modules can communicate with each other using the FOCUS messaging system. Within the FOCUS distributed architecture, the Harness module is responsible for providing user interface functionality and controlling user dialogues. The graphics functions have been developed as extensions to the Dynamic Presentation Layer and the Physical Presentation Layer [22].

A set of graphics primitives forms the basis of a set of graphical Abstract Interaction Objects (AIOs). An AIO is an object which defines contents and attributes of an object but does not specify its presentation on screen. Standard AIO definitions represent basic interaction functions, for example, selection, question and hypertext.

The AIO approach to interface specification can be seen to break down into three levels of complexity:

- interface primitives - tool kit widgets
- standard AIOs - composed of interface primitives
- interaction object/frame/group - composed of standard AIOs
The final high level interaction object is specified with a simple specification that hides, via the AIO definitions, a considerable degree of complexity at the lowest level of tool kit widgets.

The fundamental design of the FOCUS system envisaged the extension of the library of AIOs stored in the Presentation Knowledge Base. The standard AIOs can be extended by the application developer using an interactive editor.

A set of graphics primitives forms the basis of a set of graphical AIOs. In this way, graphical AIOs are an integral part of the AIO library. They extend the library by adding graphical interaction objects, rather than altering it in any fundamental way. Thus, it is possible to combine lines, circles, ellipses, boxes and text, specify suitable constraints, assign an AIO description to them and allow other modules to reference them in the normal way through the messaging system. This allows the system to create graphical interaction objects on the screen dynamically by sending an appropriate message.

The definition of a graphical interaction object can be updated in real time by the user of the system. A graphical interaction object has a set of points defined by the system developer. These points are the extents of graphical primitives, for example, the start and end points of a line, which can be manipulated by the user of the system. The Presentation Knowledge Base will keep a copy of the original library definition, until the updated object is deleted or saved.

2.3.6 Future Research Directions

2.3.6.1 Implications of User Evaluation

The demonstrator VPKSS has been evaluated by experts in vehicle conceptual design. The evaluation process was informal and based on demonstration and use. The response to the system was, however, very positive and gave rise to a number of specific issues to be
addressed in future versions. In particular, the following additional desirable attributes of such a knowledge support system for vehicle packaging were identified:

i) It was acknowledged the designer might wish, according to the needs of the design under consideration, to "bend the rules" in the knowledge base. However, and conversely, some types of rules, such as legal regulations, should be, in theory, never variable. To impose this rigidly has implications for the boundaries or constraints imposed upon the designer. Breaking rules may be creative in that by moving outside the conventional design space, new solutions may be explored. Thus, a potential conflict may exist here between optimisation and exploratory design activities. Differentiation between fixed and inviolable rules and those open to change would, in a real application in practice, need to be handled within the system explicitly.

ii) It is necessary to index the knowledge from a number of points of view in order to inspect, for example, all of the knowledge associated with a given graphical entity or all of the knowledge relevant to the relationship between the two objects. To assist the designer, explanation documents should be associated with the graphical objects.

iii) A wide range of drawing input is required and, of the physical user interface level, pen-based input for sketching is very important. This form of input device required a high degree of subtlety especially for the expression of tentative ideas.

More generally, vehicle concept design is undertaken by a team comprising designers and engineers with different skills and expertise and those factors must be addressed in any knowledge-based concurrent system approach. This issue is briefly reviewed below.

2.3.6.2 Individual and Shared Knowledge

The designer has individual design knowledge as well as the shared knowledge of design practice. The success of innovative designers depends upon the distinguishing contribution that they bring to the field. Hence, successful support systems must tackle both the
problem of the acquisition of knowledge of the individual designers and the mechanisms for the transfer of that knowledge to other designers.

Vehicle Packaging, for example, is a team task that requires the participation of people with different skills. Where multi-disciplinary teams are engaged typically in complex design tasks, it is important from the outset that there is suitable exchange of knowledge about the nature of the rules each member is using, modifying or creating. One approach is to provide a documentation facility that automatically creates a hypertext link to a standard format explanation file for each rule. In this way information can be gathered in a standard, agreed format. Thus, as the user creates a new rule, the system will add the rule title to the list and create a hypertext link to a new file, all automatically. The system can create suitable file names, links, etc and maintain the system. The user of such a system would be presented with a text (or hypertext) file to explain the rule and to record any other relevant information about it. Other members of the team, when encountering an unfamiliar rule, could directly access the explanation document. Maintenance is, for example, considerably enhanced by this facility. However, the control of change, familiar in the database world, becomes a significant issue.

To serve the needs of a team effort, the support system must be extended to become a fully fledged multi-user system. Different solutions might be appropriate in different companies depending on the organisational culture. The technical options are well known but the degree of sensitivity with which designers and companies might hold the knowledge in the system poses additional challenges to the already significant technical ones. A private design knowledge base might contain individual design strategies that form part of the design knowledge of the experienced designer. Shared access to standard domain knowledge, organisational constraints, legal regulations, existing product models could be provided as part of the shared design space. Some of the issues implementing such facilities discussed by Jones et al. [25] and in Edmonds et al. [15].

One specific issue that must be addressed is the heterogeneous character of the participants in the design team. The specific languages of interaction used (typically graphical) may
well, therefore, vary from team member to team member. The approach being investigated by the authors uses restricted natural language as the commonly available output. Thus, as a designer manipulates the graphical interface and the internal formal knowledge representations are formed, a translator program makes a direct restricted English equivalent of the knowledge available, when required. This description provides an interpretation of designer actions that are not normally readily accessible to other members of the team.

2.3.7 Conclusions

An approach to the provision of support to conceptual design by means of an interactive Knowledge Support System was described in this paper. This was demonstrated in an architecture and a system applied to the task of concept vehicle design. Our aim is to provide knowledge that will support the designer during the design process itself. The knowledge is used to support the exploration of possible design solutions by providing generative and evaluative capabilities in the system. Graphical interactive techniques allow the designer to interact with the domain knowledge in specific concept vehicle design terms. Future developments were discussed, in particular, the extension of the system support to a multi-user platform to support team working. An issue that is being addressed in our current research is how to support different participants in the design team.


The graphical interaction work has been developed further using the World Wide Web: a short account of this appears in 'A method for graphical input on the WWW' by Parks and Edmonds, In Proceedings of CHI '97.

Acknowledgements

We wish to thank the members of the Vehicle Concepts Department of Lotus Engineering, U.K. for their support and encouragement. The work was partly funded by the Engineering and Physical Sciences Research Council, Contract No. GR/J41680.
References


3.1 SUPPORTING THE CREATIVE USER: A CRITERIA-BASED APPROACH TO INTERACTION DESIGN


The paper is concerned with the design of interactive systems for creative users. It draws upon results from design research and studies of creativity. The role of criteria in design is discussed and a criteria-based modelling approach to interactive system design is proposed.

Keywords: Design Research, Interaction Designers, Criteria-based Models

3.1.1 Introduction

In this paper, a case for adopting criteria-based models that support the designer of computer systems for creative tasks is made. The criteria-based model expresses criteria that may be used to evaluate the design as opposed to task modelling, a representation form from which one might hope to deduce the design. One aim of adopting the criteria-based modelling approach is to re-orientate the way we look at requirements at present: for example, whether they match the cognitive characteristics of the user in a given situation should be an important consideration.

In the context of creative work, the interaction with a computer system is more significant in terms of how users' thinking is influenced, rather than in terms of the specific output, or answers, that the computer produces, as in a conventional expert system, or indeed, a standard spreadsheet or word processor. The situation envisaged is, first, one where users are expert in their field: the term 'knowledge workers' applies [1]. Second, the user's goals are multi-dimensional and tasks may be thought of as ill-defined, or even 'wicked' and ones which require the generation of new solutions. Such tasks could be, for example, early conceptual design where a designer explores alternative design concepts for a problem with a vague specification.

The paper begins with a scenario account of the target area under consideration where the interaction designer or design team is charged with designing a support system for creative
tasks to be carried out by a knowledge worker. The lessons to be drawn from design research are then reviewed briefly and some key elements of designer characteristics, those of solution-led design and constraint driven design, identified. The implications for interaction design are then considered. From this point, the discussion turns to the main point of the paper, the nature of criteria-based modelling and why it is appropriate for helping the design of creativity enhancing systems. In the final discussion section we consider some of the implications for task analysis.

3.1.1.1 The Scenario

Figure 1 describes the main scenario under consideration. An interaction designer is charged with the design of a computer support system for a creative user, in this case a knowledge worker. We assert that such interaction designers must be creative in themselves. In order to inform that process the designer has access to a model of the user and the intended tasks. The question is, what form of model is most helpful to the interaction designer bearing in mind the nature of the design process? The situation is one in which the interaction designer has some model of what is to be achieved by the computer system being designed and, in particular, how the knowledge worker will interact with and benefit from it. A key point is that, in the type of work envisaged, we cannot assume that the user performs tasks in closely ordered, sequential and predictable ways.

Figure 1. The Scenario
In our research, we have investigated examples of creative rather than routine tasks [2,3]. In creativity research, it is recognised that knowledge intensive tasks are critical components of creative work, e.g. Perkins [4]. In relation to the interaction designer, we take the view that the activity of computer system design may have much in common with the design of other artefacts and try to learn from what is known about design as a general discipline. Design research has a considerable literature from which it is possible to draw upon results of design practice studies that are relevant to the state of system design from an HCI perspective. Designing interactive systems for use in many task and application domains is a prime example of design in practice.

3.1.2 Design Research

There has been considerable research into how designers carry out design activities. In both product design and software design, characteristics in common have been identified: in particular, design as an hierarchically organised planned activity versus design as an opportunistically driven mix of top-down and bottom up strategies has been explored in a number of empirical studies, e.g. Guindon [5] and Ullman [6]. The results indicate that the factors that differentiate between design situations are complexity of the problem domain (e.g. aircraft or vehicle design compared to simple products), whether the problem is ill-defined or well-understood, the experience and skill of the designers and the operational conditions of the total activity in a given organisational context.

Earlier work on the organisation of software design activities showed that notions of a predefined sequence were misguided and that, not only is design characterised as a loosely structured process, but that designers are able to handle different levels of the abstraction at the same time. Pennington et al [7] examined software design, particularly object-oriented design, in relation to previous work and confirmed earlier findings. A particular point made was that there were two critical factors, those of designer experience and movement between levels of abstraction. The experienced designer is more inclined to change level frequently although other factors also apply.
3.1.2.1 Using Constraints to Drive Design

From the results of empirical studies of design thinking certain characteristics of design that are relevant to this discussion have been identified. In particular, design is often solution-led, in that early on the designer proposes solutions in order to better understand the problem [8]. Thus previous, prototypical designs are needed and re-use is a significant activity [9]. Secondly, however, it is seen that the iterations resulting from the solution-led approach are largely driven by the perceived constraints upon the design goal. These constraints (be they requirements, legal issues, stylistic preferences or whatever) generate the criteria against which each candidate solution is evaluated and the next iteration initiated.

Designers impose additional constraints that narrow the solution space and help generate concepts. Solution conjectures are used early on to narrow the solution space and all imposed strong constraints or narrow objectives on the problem to help generate the 'early solution concept'- called 'primary generators' by Darke [10]. Factors identified are:-

- Critical parts of the design in terms of knowledge or functionality are identified first as the initial focus of the conceptual design.

- Designers change goals and add constraints during the design process.

  The adjustment of goals and constraints is part of the designer's learning experience in solving ill-defined problems.

- Designers generate solutions using self-imposed constraints that are necessary to resolve the problem. Initial design concepts may be based upon invalid premises which may prove difficult to resolve. However, designers are resistant to changing those initial concepts and, instead, exercise considerable freedom in changing the goals. The creative designer's particular skill can be to surmount this and move on to the emergence of radical ideas from those early solutions.

- Designers usually treat problems as ill-defined even where they could be considered to be well-defined. This is contrary to the aim of some design research which is to
encourage systematic well-defined ways of approaching problems. The fact seems to be that designers often prefer the flexibility in relation to constraints that an ill-defined problem affords.

The key element resulting from these design studies that we wish to stress is the importance of a particular mode of working. This can be characterised as:

- hypothesize solution
- apply design criteria
- reconsider solution
- reconsider criteria
- iterate

This implies the need for two things: first, a library of potential solutions, such as successful past examples, and, second, a set of evolving criteria, such as might be defined from requirements. The implication that we address is the need to express needs, particularly the user's task needs, in terms of criteria. This implication may be contrasted with sequence-based prescriptive models of the intended user actions.

### 3.1.3 Interaction Design

This section discusses some of the implications of the above discussion for interaction designers who are working on systems of whom the end users are knowledge workers. We restrict our case to one in which the system being designed is intended for tasks that are relatively complex and hard to characterize.

There have been a number of studies in engineering, industrial and software design that argue for computer support tools that take proper account of the real way that designers design. Requirements for support systems have been proposed and implemented as demonstration systems. However, the extent to which fully-fledged commercially available systems have been based upon the results of empirical studies of designers is poor. Guindon [11] however, proposed and demonstrated, in the DesignVision system, requirements for support tools for early software design derived from empirical studies. An
analysis of the implications of such studies leads to a set of criteria for design support across different domains [12].

In the studies referred to earlier, a key point is that the precise designer activities cannot be predicted. However, certain characteristics of those activities are clear and must be taken account of. Design is not deductive. Hence any model of user tasks that supports it can only frame or scope possible solutions. The question is, what is the best way to provide such a scope? We have considered design beyond computing and noted that constraints or, perhaps more positively, criteria, help. No solution is implied by a criteria-based model. It must be seen as a tool to be used by a skilled designer of interactive systems rather that a method that can guarantee the generation of good solutions.

It is interesting to refer to the creativity literature at this point. Boden [13] suggests that changing a constraint might be at the core of creative thinking. Therefore, having the criteria explicitly available for consideration and, hence, modification by the designer of the computer system might encourage innovative solutions.

### 3.1.4 Criteria-Based Models

A model is an abstract representation of a concrete entity and therefore, does not describe everything that is relevant to the eventual realization of a specification in physical form. The items or entities it does include, or are omitted, depend upon the purpose for which the (abstract) design model is intended to be used: e.g. in respect of a research paper, the model could be an abstract of the argument or it could be the page length and format details expressed as a style sheet, depending on the intended audience and purpose.

The particular representation chosen for the model may determine the type of design result e.g. flow charts are very appropriate for representing a sequence of events with branching operations. It is necessary to ask the question for what purpose is the model being built? For example, in order to control and monitor actions and outputs, versions of a set of drawings, a design rationale or teaching materials etc., a different style of model is likely to
be required. The representation methods should be commensurate with the purpose and this must be explicit. Representation by flow chart implies a particular style of computer support because the processes involved are sequenced. So can one assume that the support should be procedural and modal? The evidence is that it is not the preferred practice of many designers, especially the experienced and innovative ones.

Consider the notion of providing support to the designer by the deployment of a model. This implies the making of a choice of what to include, because a model is an abstraction from the facts of the case. For the design of user interfaces to computer support systems, the model of the users' needs defines the boundaries of the design space of the computer support system. This is often done by using constraints (e.g. requirements defined by the characteristics of the cognitive processes that are basic to human capability such as working memory, attention span, etc.).

The application of such constraints does not in itself guarantee a "good" system design. However, it helps to focus attention during the software design activity on those design options that are most likely to succeed. In other words, conforming to constraints provides necessary, rather than sufficient, conditions for success. Now, constraints can be used to establish a set of evaluation criteria that are based upon what humans can be expected to achieve.

Portillo and Dohr [14] consider the role of criteria in the design process and argue that criteria can integrate process and structure. They assert that, by contrast, sequential models of design are not adequate; "Attempts that prescribe a single linear-sequential model for all situations appear naive in today's design world". They also review the literature on criteria and related concepts, such as constraints, in design and report on a study of the use of criteria by expert designers. The design domain of the study was colour, although only about half of the subjects specialised in that aspect of design.
Each designer in the study provided an in-depth description of a single project from which three expert judges independently extracted a total of 107 criteria that were employed. From this set the experts produced five categories of criteria by functions, listed below with example relevant issues:

<table>
<thead>
<tr>
<th>CRITERIA</th>
<th>EXAMPLE ISSUES</th>
</tr>
</thead>
<tbody>
<tr>
<td>behavioural</td>
<td>user action needs</td>
</tr>
<tr>
<td>compositional</td>
<td>aesthetics</td>
</tr>
<tr>
<td>symbolic</td>
<td>company image</td>
</tr>
<tr>
<td>preferential</td>
<td>market trends</td>
</tr>
<tr>
<td>pragmatic</td>
<td>cost</td>
</tr>
</tbody>
</table>

The extent of use of criteria in each category may vary from design task to design task and in relation to expertise. For example the colour experts used more compositional criteria than the non-experts when considering the colour task. It is interesting to note that each of the categories listed above matter in the case of software, although the attention that they receive is quite varied. The most significant observation in our context, however, is that the explicit consideration of criteria is observed to be a significant aspect of design.

Our work has focused on behavioural and preferential criteria, the latter being particularly driven by the need to take individual differences into account. Further reflection and analysis indicates that two other candidate classes of criteria are relevant. Contextual, or situational, criteria have proven to be significant and, particularly in relation to complex computer support, performance criteria are important. An example of a contextual criterion would be the need for the system to be operable within an engineering workshop. An example of a performance criterion would be that the system's response time is never greater than two seconds. In relation to the last point, the ease or difficulty in providing particular performance in an interactive system is clearly a factor that has a strong influence on the design process.
In the light of this, we consider criteria-based models with more categories than those listed by Portillo and Dohr [14] as indicated in figure 2. The criteria may be viewed as defining the boundary conditions of the support system design. Some examples of behavioural criteria that have been developed in our studies are outlined below. These provide a particular characterisation of the working pattern of the knowledge workers who were the subjects.

Figure 2: The Criteria Types

Some examples of behavioural criteria identified from previous studies are summarised below [2,15]. The examples of criteria are:

The system must enable the user to :-

1. Adopt an holistic perspective on the task or problem under consideration.
To achieve this, the user should have access, for example, to overviews, multiple views and alternative representations of the data in respect of the developing solution.

2. Keep several channels of exploration open in parallel.

To achieve this, the user should be able to alternate between tasks freely, fluently and quickly. This implies having simultaneous access to different forms of visual data and methods for knowledge interaction.

3. Explore and evaluate existing design knowledge in relation to other heterogeneous knowledge sources, generate and evaluate new concepts and apply constraints as appropriate.

For the user to achieve a truly knowledge intensive form of interaction, knowledge systems design requires certain key features as follows:-

- Knowledge Modification and Evolution

Accessible and powerful methods for modifying the knowledge base should be provided so that the user can refine existing rules and add new rules incrementally at any time during the on-going tasks.

- Evaluation of Evolving Knowledge

Support for evaluation of the evolving knowledge in progress should be provided. The user must be able to ask why or why not about the results of any request for evaluation and receive an explanation that is expressed in domain specific terms.

- Comparative Evaluation of Knowledge

Support for comparative evaluation of the knowledge in the system should be available. The user should be able to create and evaluate selected rule sets with different variables against identical criteria allowing immediate comparisons of their performance relative to each other and to a standard rule base.
The studies also showed that attempts to describe the performance of knowledge intensive tasks using sequence-based methods have proved impossible as counter examples of any postulated ordering could always be found. It would be quite false, however, to conclude that the work was done in a 'chaotic' manner. In fact, the tasks were carried out with considerable determination and reflection. Thus, whilst predictable sequencing was not observed, regularities in behavioural patterns were.

3.1.5 Discussion
Benyon [16] characterises task models as using hierarchical representations of tasks, providing sequencing information and identifying the objects and actions relevant to the user. Whilst his view has been disputed by Diaper and Addison [17] the argument is about the broader issues and not the core of Benyon's view. We can see that an hierarchical decomposition is only helpful in this context if it is predictable. The work reported in this paper suggests that for support of creative designers of systems for creative users, such predictability is not to be presumed. We must also note that the sequencing of activities is equally unpredictable. In relation to the objects of concern, we have seen that it is precisely in this respect that innovation can occur: the designer re-formulates the objects of concern during the design process.

The crucial point is that Task Analysis presumes that we know in advance what the user might want to do and how, in some sense, it might be done. Payne and Green [18] scope the various approaches but do not include cases where the task is very high level and no pre-decomposition is available. For example, the primary motivations of the knowledge worker, for example, "I want a bicycle that helps me go faster" or a systems view of a product design i.e. not just the car but its role in the transport system. The criteria-based approach, on the other hand, does not make the same assumptions and, as we have seen, is able to express more rigorous demands of the system. In that context it is important to emphasize that the expert designer will review the criteria during the design process and that significant design steps are often associated with criteria modification.
For future work, it is necessary to note that complex design tasks, by their very nature, are designed in teams both large and small, whether it be a new automobile or a software system and, thus, studies of the designer in isolation from others and carrying out small scale tasks can only realistically yield limited information about real design behaviour and outcomes. The team should be the focus of attention. Thus the sharing of criteria, the resolution of conflicts between criteria and the management of criteria model development are issues for future research.

References


3.2 COGNITIVE MODELLING OF CREATIVE KNOWLEDGE WORK FOR INTERACTION DESIGN CRITERIA


Abstract: In this paper, research into creativity and knowledge work are described and the concept of creative knowledge work is introduced. Building on an earlier criteria-based approach to interactive systems design support, the paper extends it to the area of creative knowledge work. The cognitive modelling of creative knowledge work is applied to deriving criteria for evaluation. The evaluation is carried out in relation to the different levels of system design. Examples of how the criteria can be applied are described.

1. Introduction

Three areas of research are brought together in this paper:- creativity, knowledge work and interactive computer systems design. The first two areas are combined to support the third. A distinction between creative work and knowledge work is made and the concept of the user as a creative knowledge worker is introduced.

Creativity does not need to involve knowledge work and knowledge work does not have to involve creativity. Knowledge work involves the assimilation of existing knowledge and its interpretation for the benefit of others. The knowledge worker’s knowledge is a continually evolving body of expertise but this is not necessarily creative in itself. However, where the knowledge work involves the generation and evaluation of new ideas, solutions, artefacts etc., this constitutes creative knowledge work.

Creative knowledge work conceived in this way, relies heavily upon the outcome or product of the process as the ‘benchmark’. However, the research was more fundamentally concerned with understanding the creative processes involved. This was driven by the need to inform computer system design in respect of the main goal of the research that of informing the design of interactive computer systems for supporting creative knowledge work.
A criteria-based model for interaction design support was described in Candy and Edmonds (1997). This paper extends the criteria-based approach to the area of creative knowledge work. In particular, the cognitive modelling of creative knowledge work is applied to deriving criteria for evaluating the interaction design. For the purposes of the interactive systems designer, the cognitive model defines the boundaries of the design space of the computer support system. In order to transform the cognitive model into a form that can be applied in system design, a set of criteria may be derived. These criteria represent a set of requirements of the computer support system to be designed. In order to carry out evaluation in terms of whether the system meets the cognitive requirements of the creative knowledge worker, it is necessary to specify what those criteria are, and to which parts of the system design they should be applied. The criteria that are cited are drawn from a larger set which have been identified from studies across different domains (Candy, 1998). It is recognised that the context of the users, their tasks and technical and physical environments must be considered when designing any computer support environment. The evaluation of the system is carried out in relation to the different levels of criteria and their relationship to the particular aspect of the creative cognitive processes to which they apply.

2. Creativity and Knowledge Work

In this section, research into creativity and knowledge work are described and the concept of creative knowledge work is introduced.

2.1. CONCEPTS OF CREATIVITY

Whether or not creativity can be understood or characterised in a reliable way remains a research issue. Creativity has been defined at many different levels and from various perspectives but, as yet, a fully coherent model of its multi-dimensional aspects has yet to be achieved. The tension between the popular concepts of what creativity means and the scholarly quest for an accurate, generalisable description is evident. The word itself is very widely used today and has become something of a catchall to describe any novel idea or product. Definitions and examples of creative process and creative outcomes are discussed in Boden, 1990; Candy, 1992, 1997; Partridge and Rowe, 1994.
Creativity may characterised as a *process* towards achieving an outcome that is recognised to be innovative. This characterisation goes beyond ‘everyday’ creativity which is personal to the individual concerned and does not necessarily lead to publicly scrutinised outcomes. Conceiving new ideas and making artefacts by any individual may indeed be creative to that person, but the outcomes from personal creative acts is not usually valued as such by others. Boden’s distinction between ‘P’ (psychological) and ‘H’ (historical) creative is relevant here (Boden, 1990). In this author’s view, a further distinction is needed within ‘H’ creative between *exceptional* and *outstanding* creativity. The outcomes of creative work that are exceptional may be evaluated (and valued) by others, usually the domain experts, but they not necessarily recognised as such outside that knowledgeable group. Outstanding creativity is that which has stood the test of time and has become recognised beyond the specialist community.

In order to provide focus in creativity research, it is necessary to frame the area under consideration. The creative process of certain individuals, as studied by the author, did lead to exceptional outcomes that were evaluated by peer groups. However, this does not mean that those outcomes will be outstanding in historical terms. For that to exist, the outcomes must be established over a very much longer term by a process of evaluation and comparison with other ideas and artefacts.

Creativity defined in these ways, relies heavily upon the outcome or product of the creative process as the ‘benchmark’. However, whilst the subjects were identified and selected on the basis of their exceptional results, the research was more fundamentally concerned with understanding the creative processes involved. This is driven by the need to inform computer system design in respect of how to provide support to human creativity.

2.2. KNOWLEDGE WORK

The role of knowledge in creativity is a key aspect of the work described. Creative knowledge work involves applying and developing knowledge that is both general purpose and domain specific. The knowledge worker was first identified by Drucker (1969) who
was using the term to distinguish those people who used ideas and concepts in their work from those who used mainly clerical or manual skills.

Kidd extended that definition in the light of a number of case studies and with reference to the implications for computer support (Kidd, 1994). In Kidd's view, what distinguishes knowledge workers from others is that the knowledge work is their primary motivation. Their key role is that they are skilled in assimilating existing knowledge and transforming it for the benefit of their colleagues and organisations. The distinctive features of knowledge workers is that they are, in themselves changed by the kind of work they do and their knowledge is a continually evolving body of expertise.

Knowledge intensive tasks are critical components of creative work. The development and use of knowledge as an *active* process, rather than the passive information transfer of the traditional pedagogic mode, is proposed in Perkins (1986). This approach characterises knowledge as something that is *constructed* by human enquiry and, by implication, is likely to be continually evolving as it is applied and developed in response to new circumstances. This latter characterisation of knowledge is particularly relevant to creativity and is taken in this paper.

2.3. CREATIVE KNOWLEDGE WORK

Where knowledge work involves the generation and evaluation of ideas, solutions, products or artefacts that are considered to be creative in the exceptional sense (as distinct from everyday, personal creative work), this may be characterised as *creative knowledge work*. In effect, creative knowledge work combines creativity and knowledge intensive tasks in a dynamic ongoing process.

Creative knowledge work involves a process whereby the individual draws upon a heterogeneous set of knowledge sources and then transforms that knowledge into new forms. Knowledge about domain entities, whether they take the form of visual shapes, objects, parts, complex products or textual or analytical/numerical data, plays a critical role in the development of the task in hand. But domain knowledge in itself is insufficient to
inform the exceptional creative process and needs to be accompanied by context and strategic knowledge.

People who may be characterised as creative knowledge workers address problems which require the generation and evaluation of new solutions. Such people are continuously changing their existing assumptions in order to generate new ideas and solutions. The ability to transform existing knowledge into innovative ideas and products and to convey that new knowledge to others are key characteristics that conform to the Kidd definition referred to above.

From the studies of experts, it is possible to identify more closely those aspects of the cognitive style and working practices that mark the creative knowledge worker. The characteristics of the creative knowledge worker are relevant to any analysis of the domain that may be carried out as part of the interactive systems design process.

The author’s case studies that informed the concept of creative knowledge work were carried out with experts who had demonstrated an ability to generate new knowledge or innovative products. The primary focus of their work is that of continuous knowledge development and its transformation into new concepts and products. A principal aim of the research that underpins this paper is to bring the results of studies of creative knowledge workers to bear upon the design of interactive computer support systems. The role of the case study is discussed in the following section.

3. Case Study Method

As indicated above, the ideas presented in this paper are based upon the evidence of case studies. In this section, the place of these studies in the work and the particular approach used are described.

A case study is an empirical investigation of a specific set of events within a real-life context in which a number of factors are considered as evidence (Craig Smith, 1990, Bruce, 1993). The units that are studied and analysed may range from individual studies of
outstanding designers to histories of innovative corporate culture. It has been used extensively in action research and ethnography. It is most applicable when events are not amenable to control by the investigator and when the questions posed are open ended and multi-factored. Asking questions about why and how something took place is undertaken in order to understand the meanings of the specific instance at hand. The explanations are based upon observations of existing events or recoveries of past events. The findings from such studies do not directly generalise although it is common to compare results with other similar ones. A common use of the case study is to generate hypotheses about a wide range of events which may then be studied in single variable controlled conditions using traditional experimental methods.

Case studies provide evidence about how designing takes place in real world contexts. Empirical testing involves testing knowledge claims within a context (Argyris et al, 1985). The results of observations have a bearing upon the acceptance or rejection of a theory. The connection between the theory and the results of observation should be explicit such that people can agree them. In research into human activities, controlled laboratory conditions are not achievable without sacrificing the context that gives them meaning. Instead, there are richer layers of meanings which are imposed by the context and which are relevant to the description and interpretation of what is happening. When studies of design and creativity are carried out, the context is an important consideration. The studies may be carried out at different levels of the problem scope. This is examined from different orientations from cognitive dimensions to social dynamics and design strategies.

3.1 BOUNDARY CASE METHOD

In statistically-based experimental science, the unusual cases that are single instances and fall outside the norm are inclined to be ignored or discarded as resulting from some error or other. Multiple instances in the same context are reinforced and, therefore, more confidence can be ascribed to the results. In this approach, the characteristics of the working practices of (say) the outstanding designer, who operates in a highly individualistic way, are likely to play a very small part in the analysis of the data.
Popper's proposal was to view science as an activity in which considerable effort is spent in looking for counter examples (Popper, 1969). From this point of view, a single instance that does not match the theoretical model of events, far from being discarded, is vital information. An important aspect of this is in the consideration of an observation in a specific context. Repeated changes of context with the same phenomena occurring leads to a belief in a 'generic' conclusion. If a change of context leads to no instances of the same phenomena, then it can be assumed to be related to or dependent upon the context. As an example, consider the case of the public/private space requirements that are relevant to design support for teams and individuals. It is necessary to resolve when and where to allocate these potentially conflicting features of the system and to make them available to the users as appropriate to a given context. The interactive computer system designer needs to know that conflicting requirements exist in order to resolve them. A single conflicting instance may be seen as a signal that flexibility in the requirement exists and must be addressed as such. A model of the user domain may have entries that are based on significant instances that are important in generating the flexibility of application of the system.

Another way of looking at this issue is to say that considering the boundary conditions provides insight into what is significant about the users, tasks and the work context. For example, in perception, an understanding of illusion can be most informative in revealing the core issues (Gregory, 1974). The creative knowledge worker may be seen as existing at the boundary of the space of users in general.

3.2. CASE STUDIES AND COMPUTER SUPPORT DESIGN

A number of studies have focused on outstanding designers in their fields who are unquestionably capable of creative ideas and artefacts that can be turned into innovative products (e.g. Maccoby, 1991). Although they represented a spectrum of different domains and cultures, they exhibited similar ways of thinking and working. One factor that is evident is the limited role that computers play in the personal creativity involved. In the work of exceptionally creative people, there are, as yet, few studies of the positive impact
of computers on that process or, the converse of that, those aspects of computer interaction that impede creativity.

In respect of informing computer support design, evidence about the designerly behaviour and working practices of outstanding designers has certain characteristics as outlined below.

- Given that they are likely to be demanding of the conditions in which they carry out their work in respect of their tools, materials, resources, personal strategies, this may be used to test the limits of what is acceptable and possible for computer support.

- They are known to be ready to disregard conventional wisdom and to hold opinions on a wide range of topics and, therefore, may be expected to provide a rich source of new ideas and alternative perspectives on any issue.

- Exceptional cases are likely to provide a sharper focus on the differences between people's working practices as distinct from the commonalities. This is particularly important if the boundary conditions of the design space are to be understood. The results may provide good models of practice.

- Outstanding designers are capable of presenting their ideas and insights in a fresh, but highly organised manner and the result is often far more accessible than a single instance would suppose. Many examples do not always lead to greater value.

Visser draws a distinction between what can be concluded from evidence about so-called novice designers and experts (Visser, 1994). It can equally be claimed that the study of exceptional designers working in their normal environments is likely to yield quite different results from investigating the practice of student designers in laboratory conditions. The case for the study of creative knowledge workers rests upon the assumption that their characteristics are not unlike those of exceptional designers and, as such, they need to be addressed in interactive computer support system design.

Case study findings give rise to different outcomes which may take the form of models, requirements, criteria, strategies and methods. In the studies under consideration, they yielded different kinds of evidence about creative knowledge work which were used to test existing models. These were then modified or new models were derived in the light of that
evidence. The models, in turn, informed the requirements for creativity support systems in which knowledge evaluation and extension, visualisation and collaboration are proposed as essential ingredients of creative knowledge work.

4. Cognitive Modelling of Creative Knowledge Work

This section begins with a description of previous modelling work on which the main cognitive model is based. The characterisation of each model is drawn from a number of studies including those of the author. A cognitive model of creativity was derived from the results of a study of a scientist working with an interactive knowledge-based system and building upon other creativity models. The application of that model showed that it was not fully representative of all the critical elements that are relevant to the interactive computer systems designer. The extension of the process model of creative work into a model of creative knowledge work is then described. This model of creative knowledge work was derived by selecting the elements that applied to creative work and knowledge work and those characteristics of designer practice and computer support that are relevant to those processes.

4.1 STUDIES OF CREATIVE KNOWLEDGE WORK

The author’s case study results provided different kinds of evidence about creativity, from which models, one of creative process and the other of creative cognition, were derived. These cognitive models informed the requirements for a creativity support system in which knowledge, visualisation and collaboration are, in combination, essential ingredients of creative work (Candy, 1997).

Knowledge applied in creative work is an important contribution to the innovation involved. It is a subject that has been investigated in a small number of empirical studies. Expert knowledge may be drawn upon, evaluated and reassessed: but the process is only 'creative' when it leads to the generation of new knowledge (Candy et al, 1995, Visser, 1994,1995). Visualisation involves working with visual data such as images, drawings, sketches, diagrams, charts, graphs, graphical objects, that are specific to the domain. It takes the form of expressing ideas and concepts through sketching, annotation and
examining multiple or alternative views of the same data, all of which varies according to the domain of interest. Visualisation activities are to be found in most examples of creative work although the degree of emphasis varies considerably from domain to domain (Candy et al, 1993; Tovey, 1986).

Collaboration is an aspect of creativity that has not received much attention although the role of computer systems in collaborative work is a subject area of research in itself. In the creative work of the single individual, the need for collaboration may arise only when the innovative idea requires the use of a special technique that is outside the skill of that individual. When the boundaries of the problem space are driven into new territory by a high risk strategy, this often leads to experimentation with untried methods and materials (Candy and Edmonds, 1996).

A number of models that aim to represent the creative process include these activities (e.g. Finke et al, 1992, etc.). The characterisation of each is drawn from a number of studies including those of the author and is described in brief below.

*Exploration* involves accessing source data, comprising different types of knowledge that may be examined, assessed and interpreted in terms of the primary goals of the creative knowledge worker: e.g. addressing customer requirements, problem specifications, design briefs etc. This is an open process, possibly without observable directions. However, the thoroughness and selectivity of the activity is critical to the quality of the generative stage that follows immediately and to the subsequent iterations that take place between those stages. Having to hand a comprehensive set of knowledge sources is advantageous. Knowing where to look and how to select the knowledge is even more important. There is often rapid iteration between the Exploration and Generation Activities depending on the domain or problem area as has been noted by others (e.g. Logan and Smithers, 1993; McNeil and Edmonds, 1994).

The *Generation* of possible solutions or approaches to the brief or problem definition draws upon the results of the initial exploration. Problem formulation, as distinct from problem solving, is critical to the effectiveness of the solution space that is defined. It
draws upon a wide range of analogous cases often outside the immediate domain. This has been characterised as an ability to make remote associations (Mednick, 1962). Creativity is demonstrated by the generation of many potential solutions instead of gravitating quickly towards a single and (usually) familiar solution that is not necessarily the optimal one. The ability to consider parallel lines of thought and to select and transform the results to meet the demands of a different situation is a critically important aspect of solution generation (Lawson, 1993).

*Evaluation* involves taking the results of the generative activity and testing the candidate solutions against a set of constraints. This leads to modifying, reformulating or discarding solutions depending on the feedback. Selection of the optimal solution may involve a number of trade-offs against the constraints that are applied especially where, as is usually the case, the product is a complex one. The application of tight constraints may be considered conducive to creative solution finding and thus evaluation is a vital part of the creative process (Boden, 1990; Finke et al, 1992). Evaluation may be viewed as an all-pervasive activity from the exploration phase onwards. In particular, the application of expert knowledge in evaluation has been identified as an important aspect of successful solution finding (Candy et al, 1993; Candy and Edmonds, 1995).

### 4.2 A COGNITIVE MODEL OF CREATIVITY

A cognitive model of creativity was derived from the results of a study of a scientist working with an interactive knowledge-based system and building upon other creativity models. This particular model was derived from the results of the case study described in Candy et al, 1993 and Candy and Edmonds, 1995. In Figure 1 below, a simplified version of the model is shown.
This model represents the main activities that comprised the creative process as follows:

- generation and invention
- exploration and evaluation
- constraints and requirements

The three stages of the process model indicate the progression of the human activities in the context of the study reported. The process was a form of creative work in which changes of mode took place. Further work indicated this to be insufficient to be fully representative of all the critical elements that are relevant to the interaction designer. The extension of the process model of creative work into a model of creative knowledge work is described in section 4.3 below.

4.3 A MODEL OF CREATIVE KNOWLEDGE WORK

A model of Creative Knowledge Work may be derived by selecting the elements that apply to creative design and knowledge work and those characteristics of designer practice and computer support that are relevant to those processes. In figure 2 below, the creative process is shown as a set of Activities: Exploration, Generation, Evaluation. These Activities are combined with a set of Contributors which feed into the exploratory and evaluation...
activities and arise out of the generative activities. This comprises Creative Knowledge Work.

*Contributors* are inputs and outputs to the activities. They take the form of requirements lists, product models, organisational principles and working practices, targets or criteria, design layouts, mechanical arrangements and knowledge resources. Knowledge resources comprise a heterogeneous set of contributors to the activities. They may be classified according to the domain of origin or application, the context of use and the way they are used by the actors, whether human or computer-based.

Contributors include items that represent computer files, modules, knowledge bases etc. Other contributors are derived or mediated by designers or design teams from computer-based or paper based data and information. As inputs and outputs to the creative process, the Contributors are continually undergoing transformation as the designer develops candidate solutions to meet the remit or brief. The progression is shown with iterations between the activities indicating the type of process it is. Each Activity may be decomposed into a set of tasks: the examples given are ascribed to a particular Activity but are not exclusively so.

Figure 2 below illustrates the Activities and Contributors that comprise Creative Knowledge Work. The relationship between each sub-activity is shown and the relative strength of the interaction between them. The Contributors to the Activities are expressed as different types of knowledge.
When the Activities are combined with Contributors (expressed as different types of data, information and knowledge), this comprises Knowledge Work (see Figure 3 below). In the example here, Contributors are categorised as Domain Knowledge, Context Knowledge and Strategic Knowledge. The Contributors are both inputs to and outputs of creative knowledge work and form an integral part of the evolving design. As such, they can be expected to be subject to modification.

**ACTIVITIES**

**Figure 2. Creative Activities with Contributors**

**Figure 3: Creative Knowledge Work**
The Knowledge Contributors to the creative design processes may be classified according to the domain of origin or application, the context of use and the strategies used by the actors.

*Domain Knowledge* is specialist design knowledge that applies to a particular product area or design field. It may take the form of visual, textual, numerical data and information, design cases or products, specialised procedures and methods and principles from contributing disciplines.

*Context Knowledge* is knowledge that affects the way the domain knowledge is applied. It may take the form of statutory regulations, organisational, in-house rules, macro and micro economic factors, market trends, cost factors, manufacturing processes. This type of knowledge is often outside the control of the domain specialists and is subject to continual and usually unpredictable changes.

*Strategic Knowledge* is knowledge about knowledge and how and when to apply it. It may take the form of an individual's personal selection procedure, a team decision making exercise, a corporation's marketing strategy, or knowing how to maximise company in house expert knowledge and introducing cost effective procedures. In Creative Knowledge Work, it is usually the mark of the experienced designer who has a distinctive ability to identify the critical next steps to take in order to overcome a difficulty and whose actions or guidance are likely to achieve high success rates. (Hori, 1997).

Knowledge categories are important for structuring knowledge to be held within a computer system. The area is a significant research field in itself and can only be addressed in brief in this paper.

An understanding of Creative Knowledge Work may be brought to bear on the design of interactive computer systems by modelling the key elements and their characteristics and assigning criteria for evaluation. The interactive system designer requires both
representations of the users, tasks and work context (domain model) and representations for the design of the system (computational model). In both 'design for' and 'design of' representations there will different forms and notations: e.g. work flowcharts, user models, functional hierarchies, knowledge-based structures, rules, user interface dialogue designs (Candy, 1997). Thus, interactive computer systems designers need to both understand what the user's needs and requirements are, and be able to represent their own understanding at different points in the process. Representations of users, tasks and information flow are used to mediate systems design. The representations needed will vary according to the context of use and that part of the system design process to which they are applied.

Before describing the criteria modelling approach, some issues concerning interactive systems design and the methods and techniques available are discussed in the next section.

5. Modelling for Interactive Computer Systems Design

In this section, the design of interactive systems is placed within general definitions of design and the need for appropriate methods to support that process. The criteria modelling approach is then introduced.

Design as a general concept refers to the processes of professional practice, including the ideas and artefacts that result. Design may be defined even more generically as comprising activities that lead to some kind of construct. In this sense, all professional work may be said to include acts of designing. Perkins (1986) defines it as "the human endeavour of shaping objects to a purpose.". Design also refers to the process of developing a product and the various representations (models) of the product developed during the design process.

Designing interactive systems is one example within the general area of design. In the context of software design, Winograd (1996) characterises it as: "Whenever objects are created for people to use, design is pervasive. Design may be done methodically or offhandedly, consciously or accidentally. But when people create software - or any other
product - decisions are made and objects are constructed that carry with them an intention of what those objects will do and how they will be perceived and used”. He goes on to question the education of computer professionals for its concentration on computational mechanisms and engineering methods. Whilst it is essential that software is reliable and robust, it is a single-sided focus that emphasises construction and function to the exclusion of the way the system works in the user context.

Lawson provides a characterisation of design in architecture, engineering and urban design and questions whether models of the design process can be generic or field independent. He argues that design requires the selection of appropriate knowledge which dependent upon the ideas of the individual designers for solving particular problems. He proposes the notion of design as a ‘knowledge-rich’ process which requires more knowledge than appears in the initial brief or problem specification. The designer selects from a store of information and expert strategies which may be applied to many different design briefs (Lawson, 1994).

The interactive systems design process includes the exploration of the context into which the interactive computer system is to be placed (domain modelling) as well as a human factors analysis (cognitive modelling). The model of the users' needs defines the boundaries of the design space of the computer support system. Design at the conceptual and implementation level includes operator modelling and system modelling. A model is an abstract representation of a concrete entity and therefore, does not describe everything that is relevant to the eventual realisation of a specification in physical form. The specific representation chosen for the model may determine the type of design result e.g. flow charts are very appropriate for representing a sequence of events with branching operations.

Taking Lawson's view of design as a knowledge rich process where the designer applies expertise to generate creative solutions, this draws attention to the potential disadvantage of a modelling approach that aims to prescribe very specific solutions. If the domain modelling, whether by task analysis or some other method, leads to ‘recipe-like’ solutions,
this is likely to have the effect of reducing the designer's scope for creativity in interactive systems design.

The methods and techniques available to the interactive systems designer range from natural language to formal and semi-formal models. The selection of suitable representations is necessary for the exploration, generation and evaluation of ideas and decisions within the design team and with users. A number of approaches have been developed in the field of Human Computer Interaction (HCI) and software engineering for the design and implementation of interactive systems. It is clear, however, that there is a considerable gap between the domain modelling that informs the overall system requirements for design and the software design itself which is expressed as computational formalisms. The bridging of the gap between a user-centred focus such as that of task analysis and a system-centred focus such as functional design or knowledge-base design has yet to be successfully achieved. Representations that capture the work practices of a given situation may play a role in informing the computer system designer about the existing processes but, nevertheless, do not necessarily provide the best means of modelling the computational system to be developed. That will depend on the particular circumstances of the development project. If, for example, the aim is to automate parts of the process and replace some human-driven aspects of it, the representation will need to model explicitly the allocation of tasks between humans and machines. To do that effectively, the information gathered must discriminate between tasks best done by human and those best achieved by machine. This is difficult to achieve and the plethora of methods that have been developed is testimony to the depth and breadth of the search for solutions (see Benyon and Palanque, 1996).

In a previous paper, a criteria-based approach for supporting interactive computer systems design was proposed (Candy and Edmonds, 1997). In this approach, the model expresses criteria for evaluating the system design, as distinct from deducing it from a task or domain model. The criteria-based approach is intended to be a tool to be used by a designer of interactive systems rather than a prescriptive method that purports to guarantee good designs. The approach assumes a system life cycle model in which evaluation is placed at
the centre of that process. A set of criteria categories were identified and applied: e.g. behavioural: (user actions), compositional (aesthetics), preferential (market trends). The initial work focused on behavioural and preferential criteria and later two additional classes of criteria were added: contextual/situational and performance criteria.

The extension of that criteria-based approach, as described in the following sections, is based upon the notion of the boundary condition as described in section 3.1. above. This involves considering the widest possible range of domain characteristics in order to identify the boundary conditions of the design space within which designer needs to operate. These conditions enable the derivation of a set of criteria against which the design of the computer system may be evaluated and is described in section 6. The modelling approach uses a representation of the domain process for which criteria for creative knowledge work are assigned and set against levels of system design as described in section 7 below.

6. Criteria Categories for Interaction Design

For each element of the Creative Knowledge Work Model, a set of criteria have been identified. The examples shown are drawn from a larger set described in Candy, 1998.

For each Activity, as represented in the model, Creative Knowledge Work implies a basic set of framing activities. The criteria cited below apply to the specified Activity but not exclusively.

6.1 PROCESS ACTIVITIES

6.1.1 Exploration

For the Exploration Activity, a support system should allow the user to:

- *Explore* the work space (design brief, problem specification...) by examination, analysis, interpretation and selection.

- *Access* domain, context and strategic knowledge at all times during the Activity
The support system should allow the user to have immediate access to a range of functions and knowledge resources: i.e. source data such as designs, sketches, scheme drawings, graphs, charts, histories etc. This requires access to:

- source data modules and applications
- browsing facilities with search mechanisms
- overviews of source data
- different levels of representation of source data
- alternative views (representations) of source data
- switching facilities between views and levels

The interaction techniques should provide:

- minimal actions to select support functions
- display facilities for viewing and selection
- techniques for rapid transfer between display items

6.1.2. Generation

For Generation, a support system should allow the user to:

- Generate several candidate solutions and refine and transform solutions selected giving rise to new knowledge
- Apply domain, context and strategic knowledge

The support system should enable the user to:

- generate candidate solutions drawing on analogies with other examples
- compare candidate solutions with other similar examples
- reconstruct existing examples as new solutions
- generate and refine solutions in personal work space
- have access to shared spaces for collaborative work
- develop solutions in private space and transfer to shared space

The interaction techniques should provide:

- facilities for comparing objects with similar items
• graphical techniques for annotating the item under analysis
• dialogue paths for moving back and forth through the development of a task
• methods for suspending a task at any point and reverting to it.

6.1.3. Evaluation
For Evaluation, a support system should allow the user to:
• Evaluate candidate solutions or designs, analyse feedback, modify, reformulate or discard depending on results
• Apply domain, context and strategic knowledge

The support system should allow the user to:
• access rules in knowledge base and modify where appropriate
• evaluate alternative solutions against domain specific constraints
• obtain negative and positive results of tests
• access different types of knowledge in system

The interaction techniques should provide:
• graphical techniques for interacting with knowledge in system
• domain specific objects and terms of interaction
• access to rules and visual data simultaneously during evaluation
• overviews of solutions during reduction of alternatives

6.3 KNOWLEDGE CONTRIBUTORS
Criteria that specifically address the role of knowledge in creative work have been drawn from number of studies that investigated the role of knowledge in design: e.g. Gregory, 1992; Lawson, 1993, 1994; Candy et al, 1993; Visser, 1995; Bruderlin, 1996. A selection from these sources is provided below for illustration.

Computer Support Systems for Creative Knowledge Work should allow the user to:
• Draw upon heterogenous knowledge sources: case histories, existing designs, physical models, conversations with other designers, technical journals etc.
• Have access to knowledge as a set of constraints in the form of domain specific rules that can be used to test and generate design ideas as they are being developed.
• Have access to facilities for capturing knowledge dynamically as the design proceeds and restructure to personal style
• Have a plurality of representations so that new knowledge structures that emerge as a result of changes in the user's understanding can be readily incorporated into his or her activities.
• Have multiple sources of knowledge about a range of issues that may or may not be useful to the evolution of new design.
• Interact with several layers of information
• Conversations with other designers, technical journals Have access to a number of prototypical designs as well as rules about legal requirements, costs, manufacturing processes and other constraints.
• Have information about constraints and underlying assumptions of the system that are accessible during the process of developing the design.

7. Applying the Criteria to Interaction Design

For the purposes of interactive systems design support for creative knowledge work, the criteria for evaluating the system may be applied to three levels of system design.

• Interaction Configuration
• Interaction Style
• Interaction Quality

*Interaction Configuration* is defined as the user interface and the user support modules combined in a temporal structure to provide a set of functions to the user.

*Interaction Style* is the selected technique or dialogue by which the user communicates with the computer system and which shapes the user interaction behaviour.

*Interaction Quality* is defined as the quality of support to the user that the interaction configuration and interaction style provide.
Adopting a user-centred design perspective, the Interaction Configuration includes any system modules, applications and (peripheral) devices that are accessed through the user interface: these include drawing systems, statistical packages and databases, knowledge-based modules, communication and collaboration applications such as video conferencing, voice mail, email and internet browsers. Interaction Styles include any form of available technique and combinations thereof including direct manipulation, menu driven, command line, form filling and natural language.

Interaction Quality may be assessed by defining and applying performance criteria to the Interaction Configuration and choice of Interaction Style. Performance criteria may be applied to the quality of the support modules as well as to the software performance at any level. The accuracy and completeness of a set of rules which are used (for example) to generate an engineering operations layout may be measured against a set of criteria. However, this performance criteria is the province of domain experts rather than the interaction designer. On the other hand, criteria for robustness and reliability can be readily defined from software design principles and guidelines.

Inevitably, the performance criteria will be subject to contextual criteria which might be defined, for example, by whether the access required is to a global WWW site or a local online service. These criteria will be important in determining whether the Interaction Configuration meets the priorities of the users and task needs. If a particular service is required more than others, it might be necessary to provide more persistent availability than that provided by the usual caching procedures.

Performance criteria for interaction quality may be defined using metrics (MUSiC, 1993) and will be inevitably be influenced by the particular technical infrastructure of the application development.

In the example applications described in 7.1 below, criteria are applied to the different levels of the system. The criteria presented represent a set of design requirements of the computer support system to be designed. Each criterion may be defined as a high level
statement which is then decomposed into more detailed elements until it is eventually expressed as a design feature of the system. The evaluation of the system is carried out in relation to the different types of criteria and their relationship to the aspect of creative knowledge work processes to which they apply.

The interaction designer's scope includes all those components and modules of the computer system that the user is intended to interact with even where it is not necessary to construct each one from scratch. The integration of different applications and models is often necessary and where the applications are not in themselves integrated, the user interface design is expected to take account of inter-application interactions.

7.1. EXAMPLE APPLICATIONS OF CRITERIA MODELLING

In this section, three examples of applications that meet selected criteria for supporting creative knowledge work are described in brief.

*Criteria for Interaction Style*

- graphical techniques for interacting with knowledge
- domain specific objects and terms of interaction

Much work in creative design is carried out using visual data such as sketches, drawings and diagrams. A method was developed that includes manipulable graphical objects within Web pages. These objects can be used as input devices because all of the information about them, including the current values of variable elements can be sent to the knowledge-base in the server (Parks and Edmonds, 1997). For example, the user, in this case a suspension engineer, is able to move a representation of a domain object and change elements, such as line lengths. Such changes represent design changes and, as such, can be reported back to the knowledge-base and the consequences evaluated. For example, if the change had implications for another engineer, this can be reported or acted upon by the system. It is typically in such a situation that a conflict might emerge and the relevant strategic knowledge be activated (Edmonds, 1997a).
Criteria for Interaction Configuration

- generate and refine solutions in personal work space
- have access to shared spaces for collaborative work
- develop solutions in private space and transfer to shared space as required

Support for collaboration between colleagues who are working at different locations enables users to participate in a shared environment and work independently at the same time. This enables users to have an awareness of the shared artefact and to explore a particular issue in a private space. The criteria for private and shared work space were applied to interaction configuration functions.

A computer-based meeting scenario was devised and a demonstrator constructed using Web-based tools. From a user's point of view, the screen consists of three areas: the conference management area (including head and shoulders, video and sound), a shared whiteboard-like area and a private work space. The private space consists of a web browser which can display any HTML page and might, for example, be used to look at a parts catalogue. In this particular case the important pages are ones that communicate with the knowledge-base. For simple interactions, where tables etc. are normally used, form interfaces to the knowledge are used (Edmonds, 1997b).

Criteria for Interaction Configuration

- access rules in knowledge base and modify where appropriate
- evaluate alternative solutions against domain specific constraints

An example of a demonstrator system that meets certain criteria for the evaluation activity of creative knowledge work is The Vehicle Packager Knowledge Support System (VPKSS) (Candy et al, 1995). This provides support that enables the evaluation of alternative solutions. The knowledge in the system embodies sets of prototypical designs as well as rules about legal requirements, costs, manufacturing processes and other constraints. The access to knowledge as a set of constraints in the form of domain specific rules that can be used to test and generate design ideas as they are being developed.
Further work is underway in which the criteria are being developed further and applied to a number of existing creative media products: e.g. Life Forms (Calvert et al., 1993), Electronic Cocktail Napkin (Gross, 1996).

8. Conclusions

The paper has presented an approach to cognitive modelling of creative knowledge work and to interactive systems design support. Criteria for evaluating the interaction design were derived from studies of creative knowledge work. By applying criteria, the attention is focused upon the creative and knowledge intensive aspects of the users' needs. Thus, the characteristics of the creative knowledge worker are relevant to any analysis of the domain that may be carried out as part of the interactive systems design process. The work represented in this paper is concerned with applying the criteria to existing applications developed.

The validation of the model of creative knowledge work and the related criteria for computer support is an ongoing process. First, this requires the application and testing of the work in practice. Second, it requires investigations to be conducted in a broader scope of creative endeavour. The scope of this work is being extended to the domain of the arts. The goal is to support the identification of those elements that are appropriate for inclusion within future interactive computer systems for art practice.

Acknowledgments

The work was partly funded by EPSRC Grant GR/J43769. The author would like to thank the following for their contributions to the work described: the subjects of the case studies who provided their time and effort so generously; Bryan Murray, Lesley Parks and Julian Highfield for their work on the LUTCIH demonstrators and Ernest Edmonds, project leader of the MULTIK and ROPA projects.

References


4.1 COMPUTERS AND THE CURRICULUM: AN ACTION RESEARCH PERSPECTIVE


The paper raises general issues about the way computer systems are developed and relates them to education, in particular the role of teachers in research into classroom applications. The views expressed arise from research into the use of word processors in secondary schools. The broad framework of the research work undertaken has been the relationship between the introduction of computers, the perception of users and the conditions for genuine change in practice. Curriculum research is a necessary part of the process of change but it must address the practical context through those most directly involved in it. The introduction of computers across the curriculum has raised questions about their impact on learning and teaching. The key question is how teachers might be active participants in such a process. Using a simplified representation of the innovation process, issues are raised about the development and evaluation of computer systems, the context into which they are introduced, the people who use them and the interrelationship between those elements.

Keywords: conditions for change, innovation process, computer systems, action research

4.1.1 Conditions for Innovation and Change

The successful introduction of curriculum change in schools is primarily dependent upon the expertise and commitment of teachers, followed by classroom and school resources. Curriculum research which does not address the practical context and does not involve the teachers directly is unlikely to influence principles and practices and, therefore, be a stimulus to curriculum change. Views about the role research could play in education changed when it became apparent that the curriculum development projects of the sixties that aimed to alter classroom practice had not been effective (Shipman, 1985). Those projects that examined classroom activity and teacher practice were focused more towards producing innovative ideas embodied in concrete results, usually in the form of teaching packages and handbooks. A belief that high quality products would encourage the dissemination of new ideas proved to be unfounded (Stenhouse, 1980). The introduction of
computers into schools and classrooms has raised similar questions in relation to
technology-driven innovation (Gray, 1982). In particular, there is an emphasis on
developing software and hardware products for their own sakes, separate from the needs of
the curriculum and the requirements of teachers and students. That emphasis contrasts
with the small amount of field research into the value of computer use in the classroom. As
a result, there remains a lack of hard evidence about the capacity of computers to make
genuine improvements in education.

This paper discusses a number of issues which underpin its central concern, that is the
difficulty of bringing about genuine change in education. How to identify and exploit
computer technology, whether in the form of existing software and hardware or new
developments, is placed in the context of effecting change in both curriculum content and
teaching methods. It is proposed that the role of teacher-led innovation in that process is of
prime importance.

In recent times, the strategy adopted by the Microelectronics in Education Programme
(Fothergill, 1981) and the Scottish equivalent, the SMDP, for introducing computers into
the school curriculum failed to address satisfactorily the issue of how to enable the
classroom teacher to explore the potential of new technology and, hence, to influence
educational practice (Fothergill, 1986; Odor and Entwistle, 1982). Substantial sums of
money were spent on courses for teachers but there is no evidence that it was an effective
strategy. The 'cascade' notion, upon which this activity was based, whereby such teachers
pass on their new found knowledge to their colleagues does not appear to work. This
notion is a variant of the traditional approach to the dissemination of new curriculum ideas
and professional development, the in-service model, whereby teachers receive an injection
of new ideas outside the school context and from whence they are expected to transfer this
into their classroom practice. Today, in many parts of the country, despite that nationwide
effort and the commitment of many enthusiasts, the use of computers hardly impinges on
some subject areas and, where it does, is a limited experience for many children
(Watson, 1986).
The nub of the problem lies in the development of effective strategies for making computer systems, whether advanced or not, part of the normal experience of teachers and pupils alike and, more importantly, a means of increasing effectiveness in learning and life. The ability of computer use to improve people's effectiveness, whether by enhancing education or transforming the work environment, has not been established convincingly in the minds of many. There is a need for more careful consideration of the wider issues and an examination of the implications for research and development methods. It is interesting to consider approaches that are being used in commercial, business and academic research and development into new computer systems and, in particular, the rapidly expanding area of expert systems applications.

How to make 'better' computer systems is a problem which occupies researchers in the fields of human factors, psychology, computer science and engineering in both academic and industrial organisations. At present, these groups are mainly working independent of one another (Long, 1986). Few of them tackle the development of systems from the starting point of how to make them fit the current needs of people. An even more significant issue, in the longer term, is the ever changing requirements of the users as they move from infrequent usage to regular contact.

Interest in involving users in the design and development process is, however, growing. Some years ago, Enid Mumford argued for the introduction of participative computer systems design for the business world (Mumford, 1983). In the UK, British Telecom and Hewlett Packard, amongst a small but expanding group of companies, have, in the recent past, begun to place more emphasis on the involvement of potential users in the design and development process from the outset. In a survey carried out for the Alvey Directorate on current expert systems development, the importance of the users as "equal or dominant partners in the design team" was identified (Berry and Broadbent, 1987). Thus, in industry, it is being increasingly recognised that, without the direct participation of the users in the design and evaluation process, computer systems will not work effectively, will not be accepted into the work environment and, therefore, will not provide the commercial edge the companies hope for. If this situation applies in a business context, it is equally
applicable in education, where the prospective users are professional teachers with existing expertise who are unlikely to accept innovation uncritically. The opportunities for teachers to be involved in the design and development of computer systems from the outset are limited, to say the least. The educational world has little influence on the powerful commercial interests which drive and control the field. However, there is no reason why they should not be able to play a more active part in the evaluation of the existing software and hardware available to schools. More importantly, research into its application in classrooms and the concurrent effect on learning and teaching strategies could provide a strong foundation upon which such evaluation is based. Here too, the teacher's role is critical.

4.1.2 Action Research and Computer Use

One approach which can provide an appropriate framework for innovative curriculum work by teacher-researchers is action research. This is a research approach which has gained ground in education as the desire for a more effective transfer between research findings and classroom practice has emerged. Action research is a research method that aims to identify existing assumptions in practice and then develop new strategies in the light of that knowledge. The term, coined by Lewin (1947), referred to a process which was different from individual self-evaluation and which embodied the integration of practice and research. It has been deployed where the aim has been to improve teaching by means of a more reflective professional practice. It also provides a suitable framework for the introduction of new curriculum developments. The change approach is usually an iterative one beginning with a "small scale intervention in the functioning of the real world and a close examination of the effects of that intervention" (Cohen and Manion, 1981, p49). The methods include constant monitoring by participant researchers on the basis of which modifications and adjustments are made to improve the process of innovation. This involves the use of a number of data collection mechanisms (field observation notes, questionnaires, diaries, interviews, case studies,) providing a variety of perspectives in order that the feedback can be translated into modifications and directional changes. It is a means of making improvements in the current context rather than at a later time. In this
respect, it differs from more traditional experimental research which is directed towards seeking generally applicable results.

The action research approach is closely allied to the argument for teacher conducted research for which it is ideally suited. Elliott (1985) compares the action research process in educational research to scientific method in other disciplines. The process involves a clarification of a problem in which implicit assumptions are identified and made explicit from whence appropriate actions are taken to resolve the issue. This is particularly important where the process involves introducing an entirely novel approach such as computer use into the situation. The view taken by the director of the National Development Programme in Computer Aided Learning (NDPCAL), Richard Hooper, in breaking with the conventional wisdom of experimental research was a move in that direction in a large scale innovative project. Although not named as such, the intention to "proceed eclectically picking up information likely to contribute to improvement and understanding of the innovation being attempted" has the hallmark of action research (Hooper, 1977). The reason for using this method in the NDPCAL was the requirement to ensure the success of the introduction of innovation by making the process responsive and flexible and (most important) acceptable to the teachers involved. The inter-relationship of innovatory development, teacher involvement and the research aspect was central to the thinking behind this project. In the event, the arrival of microcomputers on the scene rendered many developments obsolete. However, much was learned about the implications of introducing technology into existing situations (Hooper, op cit). Today, the microcomputer system affords a better opportunity to study the impact of such innovation on a greater variety of sites outside the laboratory. In identifying and evaluating the role of computers in the curriculum, the action research approach has distinct advantages. Walker (1982) argues that, because new styles of learning are opened up by computer use, they cannot be evaluated by comparative conventional experiments. This view is supported by Somekh (1986), who points out that those learning activities promoted by microcomputer use such as group work and collaborative writing are more readily tackled by action research. The close investigation into the learning and teaching processes at work when a computer is introduced, can be carried out only by considering
the impact and effectiveness in the school and classroom context. The changes observable
by the teacher, reflecting on a number of different factors, can be related to previous
experience and assessed with a view to subsequent action. The paucity of educational
research which gives such a detailed perspective on the precise effects of introducing
computer use is regrettable. The beginning of new movements in curriculum development is
the right time to be monitoring and evaluating the situation.

4.1.3 Innovation with computers and the role of the teacher

The relationship between computer use and its effect should be seen in the context of what
makes real change in people's behaviour and actions in response to innovation. In the
introduction of computer systems into education and workplaces, there is a role for
professional research and technical expertise in providing consultation on suitable
techniques and tools for action (Anderson, 1985). However, for innovatory computer use to
succeed, in the sense of changing existing principles and practices, rather than simply
reiterating what is there in another medium, the process must involve the teachers directly
for it is they who will implement the changes inside the classrooms. Fullan (1982), referring
to change in education, concludes his analysis of what change in practice is by stating that
what determines the outcome of change is "along the three dimensions—in materials,
teaching approaches, beliefs—what people do and think". By bringing equipment into
schools first and then worrying about what to do with it, as has been the case in the UK,
the situation is one where, not only is the cart before the horse but the driver has been left
behind at the resting place down the road.

In introducing computer technology into education, it is necessary that the process begins
with an examination of existing classroom practices. This, in turn, can lead to a reappraisal
in the light of what the technology offers. In one sense, the computer use acts as a catalyst.
It may be the case that the true innovation is in what goes on inside the head of the teacher
as he or she explores the new computer tools with students. This exploratory process is
best carried out by teachers themselves with, if appropriate, the assistance of external
consultants. The notion of 'reflection-in action' (Argyris and Schon, 1974) where the
practitioner's knowledge is embedded consciously in action rather than applied to it, is a
perspective which recognises the existence of theories of action. The value of teachers identifying and exploring such theories of action lies in the insights they may derive into how to modify and improve both principles and classroom practice. In this way, within a wider programme of curriculum innovation, and with the necessary resources and support for their endeavours, teachers are more able to make new ideas a practical reality.

Developing and using computers for learning, a resource with the potential to transform educational practice, cannot be carried out effectively without consideration of the whole context into which it is to be introduced, including the current role of teachers. The temptation to promote computer use independent of the existing context is evident amongst some research workers in the information technology community. Sage and Smith (1985) argue that research into learning and information technology "should be focused on the future, rather than the present" and that the priority should be to direct research effort towards the "management of changes in educational practice". They contend that having a model of change will enable the curriculum developers to do their work effectively. However, I would question whether producing models of change is in any way likely to make changes happen where it really matters, i.e. in the minds and methods of teachers.

The necessity for research and development based upon existing practical experience and with teacher-researchers involved has been argued here. For educational practitioners, the first priority must be how to advance the curriculum, and with it teaching and learning strategies, by identifying those computer innovations which are truly beneficial. In order to do this, they need to be active participants in that change process implicated by the introduction of computer use. The discussion to follow illustrates the dynamic nature of the change process and, in particular, the participant's changing requirements and perceptions.

4.1.4 A model of innovation: from a school study in computer use

Figure 1 represents two possible models of the innovation process in a general context. The group I have dubbed the 'initiators' comprises people with a variety of perspectives on computer systems. They include computer specialists, i.e.: knowledge engineers, user
interface designers and programmers as well as human factors researchers, ergonomists and psychologists. The other group, the 'practitioners', have different goals to the initiators with respect to computer systems, in that the role of the technology is subject to the demands of the particular domain in which the individual is operating, whether that be learning to structure an essay, writing a book, designing a new product, making a medical diagnosis or teaching mathematics. According to the model of the innovation process as shown in Figure 1, in the traditional design and development process, the initiators determine requirements for computer systems independently of the practitioners. Such a model does not allow for what happens once the practitioners have begun to use the system. The initial process of learning in order to carry out a task, followed by a more extensive exploration of the available facilities, is inherently dynamic. In other words, the conceptual context or environment of the person involved will be ever changing. The implications of such a process for the development and use of computer systems are significant.

In the example to follow, the young writers are, in effect, the practitioners of Figure 1. In work with 14 to 16 year old secondary school students, it was observed that changes in writing goals and strategies occurred even where there was no adaptive response from the
system and there was minimal direct intervention from teachers (Candy, 1986). However, it should be noted that the initial changes in student behaviour were of a restricted kind in terms of true writing development.

Three stages in the early development of young writer's use of a writing aid can be identified (see Figure 2). I shall use these stages to illustrate the dynamic process implied in the model illustrated in Figure 1. The examples below come from a study, begun in 1980, into the use of computers in English teaching. Work carried out by the author in a Leicestershire secondary school was aimed at gaining understanding of the whole process of introducing computer use in the context of an English department and its curriculum. The complete study provides an example of how the introduction of computers and teacher research can be seen as joint components of the process of change. The particular example used derives from the findings about the use of the word processor amongst students aged between 14 and 16 years, initially over a one year period of close monitoring by observation and the use of keystroke logging. Later extension of the word processor into curriculum strategies for writing development was based upon the earlier study's finding that without a structured learning framework, there were observable disadvantages to writing development. (Candy, op cit).

Figure 2. Stages in early development of word processor use
The first stage in writing with a word processor is characterized by the overriding concern for neat 'copy', often to the exclusion of accuracy. This reflects a pre-occupation with the end-product of writing and a requirement for good print quality, predictable layout and consistency of form. Taking this as one's guideline, one might concentrate on producing a writing aid which placed priority on variable fonts, print size and layout capability and the best available printer on the market. An advanced, high quality typewriter perhaps? For teachers and pupils who do not have direct experience, the concept of a word processor may not extend beyond this stage. However, in the study, all the students moved, without prompting, on to the next stage.

This second stage is marked by attention to minor details of mechanical accuracy e.g. spelling, punctuation and small scale adjustments to language. The requirement for mechanical and grammatical accuracy suggests writing systems which check on these features e.g. spelling checkers or more sophisticated grammatical and syntactical systems such as the Epistle text-critiquing system (Miller et al 1982) or the Writer's Workbench (MacDonald, 1983) or the college edition (AT&T Technologies inc.). To a large extent, this is the furthest that writing aids have reached in schools, at least in the USA. However, the important point is that if the system is such that its intervention interrupts the flow of ideas or diverts the writer to low level considerations or inhibits learning by offering a solution too readily, its effect is counterproductive. In these cases, the requirement is for flexible adaptivity by the system. Edmonds (1986) suggests that the influence of the categories of user activity and the appropriateness of the system can be viewed in terms of levels of the human-computer interface. The important point is that user goals and performance levels in a particular context are always changing and, therefore, must require dynamic monitoring. The flexibility of the system is essential for appropriate adaptivity but the crucial issue is the degree to which the user exercises control over that flexibility.

In the example under discussion, if the primary goal of the writer is to write a story uninhibited by technical details then the system needs to be adaptable to take account of this. If, on the other hand, the writer is a dyslexic child whose main concern is to focus on misspellings or produce an error free account, the system response needs to be appropriate
to those distinct goals, i.e. in the first instance not to provide an automatic replacement but to identify the errors and offer a strategy to cope with them. The difficult issue is how to enable users to specify their requirements to the system without compromising the fluency and consistency of the interface. These requirements will always be changing because, in the case of writing tools, the likelihood of changes in goals and performance is high. The writing aid itself is a catalyst in that direction.

In the third stage of writing development the focus on the level of content and organisation takes place. The young writer's goals have changed in the sense that the writing aid solutions to the former pre-occupations have now been internalised and form part of his or her implicit assumptions. In a similar way, all practitioners move through this process of changing requirements as their concept of the task in hand alters because of the impact of the technological tools. Nevertheless, the computer system must be able to handle requirements at all stages. In the particular context of computer writing aids, it is necessary to recognise both the dynamic state of the individual writer's goals (including the strategies adopted according to the writing tasks) as well as the variety of writing tasks (O'Malley and Sharples, 1986). In the case of writing systems, the context of the writing, the individual's current writing strategies and the effect of the system itself on goals and actions all contribute to an ever-changing environment.

In considering these issues in relation to design of computer systems in general, an analogy can be drawn with the development of the sewing machine and its use by both casual users and those with requirements for complex refinements. The basic function of the sewing machine, i.e. to sew a seam was greatly enhanced by the introduction of motor driven models which freed both hands. Once that step was reached, the enhancements added capability to the machine that allowed for fancy work, zigzagging, button-holing etc. For many, these features are superfluous to the basic need to sew a seam quickly. If that function operates smoothly and at a flexible pace for different levels of expertise, this in itself can justify having a sewing machine. At the same time, the more demanding requirements of the expert sewer must be met without reducing the usability for the simpler needs. In the same way, computer systems must be able to match the requirements
of users who are at varying levels of experience. To take a simple example, if a running word count on a word processor affects the speed of editing and is hard for the writer to accommodate after being used to a faster response time, it is inappropriate to include it without user control. Similarly, the speed of a touch typist is such that some desk top systems cannot provide sufficiently fast response and may be rejected in spite of other attractions such as variable fonts, window management and an easy to use interface.

4.1.5 Conclusions

The importance of user involvement in computer system design and development is increasingly recognised in the commercial world. The introduction of computers into schools raises similar questions but the scope for direct involvement is more limited. Nevertheless, the involvement of teachers in the introduction and evaluation of available technology by conducting their own investigations is essential. School based action research on computer use in the curriculum is needed if sufficient understanding is to be gained about the full impact of such innovation and its implications for learning. The writing example described can be applied to a wider context. Opportunities for the exploitation of computer resources in the curriculum in the future will succeed in educational terms if teachers are able to play an active part in research and evaluation in the field. From the evidence accrued about the real impact of computer use in the classroom, it might indeed be possible to plan better programmes to support continuing dissemination and interchange of ideas and experiences amongst teachers. However, for that dissemination to be successful it depends upon the existence of documented evidence from innovatory classroom teachers.

Acknowledgements

Thanks are expressed to Professor Derek Teather of the Maths, Computing and Statistics Department, Leicester Polytechnic and Mr Tony Clarke of the Computer Studies Department, Loughborough University of Technology for their help in the preparation of this paper.
References


Writer's Workbench College Edition. AT&T Technologies Inc. P.O. Box Greensboro, NC 27420.


4.2 Twin Paths of Research and Design: Reformulating the Process


The concepts and practices described in this chapter emerged during longitudinal studies into the design, construction, evaluation and use of a complex computer support system. The primary case argued is that the derivation of useful results from research requires strategies and techniques that are responsive to the particular context. In advanced interactive systems design, because there is so much technological innovation involved that forms an integral part of the research, the clear distinction between the research and design streams is not always recognised. The approach advocated seeks to marry research and design and yet keep the goals and outcomes distinctive.

Keywords: research methodology, participatory research, technology innovation, system design, knowledge support systems

4.2.1 Introduction

The concepts and practices described in this paper emerged during longitudinal studies into the design and use of a complex computer support system [OBR 92 a&b; CAN 93]. From the initial goal of constructing a computational model of speech, a radical re-formulation of the research problem took place. That goal changed to one of designing support systems for the evolution of human knowledge. The re-orientation towards human-centred problems rather than machine-centred ones brought about a need for appropriate methods for understanding human processes from which the requirements for computer-based system design could be identified. For that purpose, traditional rationalistic research methods proved to be unsuitable and a participatory approach was adopted.

Computer support for knowledge work combines user interaction with visual data with methods of externalising and capturing domain knowledge in the system. The domain expert is provided with computer tools that support highly interactive use of visual data and, also a means of specifying the knowledge gained from studying the data graphically.
and in text form. Such systems open the notion of computer-based support to a wider group of users and domains and are termed Knowledge Support Systems (KSS) [EDM 93]. This kind of system provides support for knowledge capture and extension and is a special case of a knowledge-based system where the user is the expert and the human expertise is not assumed to be static but evolving. This also distinguishes it clearly from the notion of an Expert System which contains codified knowledge of the human expert and is intended to provide a more complete form of advice. Knowledge Support Systems are new and important indicators of the scope of future computer-based support for complex tasks. The tasks of such knowledge work are complex and the users are domain experts with novel and often loosely defined problems. This means that the requirements for design and development of such systems have a different set of priorities to those typically met by system developers. As a result, certain significant issues are highlighted and motivate this paper.

The primary case argued is that the derivation of results from research that are applicable and usable, requires strategies and techniques that are responsive to multiple issues in the particular context. Thus, participatory research, in tandem with system design, are the twin paths that are proposed for the development of complex user systems. In advanced interactive system design, because there is so much technological innovation involved that forms an integral part of the research, the clear distinction between the research and design streams is not always recognized. The approach advocated here seeks both to marry research and design and yet keep the goals and outcomes distinctive.

4.2.2 Research Approaches in Computer System Development

The scientific paradigm based upon the rationalist tradition of enquiry has dominated computer system research and development, resulting in a preoccupation with the technology itself, its performance and formal characteristics. When the human user of the system is brought into the scope of concern, the tendency has been to import the same approach: that is, in its simplest form, to contrive situations where the user and system can be observed and monitored, a limited set of variables can be manipulated and the resulting
model can be used to make predictions (see, for example [RAS 86] REA 90] [REI 81] [REI
90].

In the climate of high expectations and low returns that characterised computer system
development in the 70s, the impetus for more effective and better designed products that
meet user and organizational requirements gave the Human-Computer Interaction (HCI)
community its very raison d'être. HCI has grown rapidly since the 1970s, particularly with
the advent of personal computing and the widespread extension of computers into
everyday life. The research agenda itself has, in part, been determined by the drive to meet
the needs of a changing population of users, but also by the very character of the
disciplines that have come together in this field. Thus, HCI is multi-disciplinary and
eclectic in its concerns [GAI 84] and, for that reason, holds within it contending forces for
the central agenda.

The selection of lines of scientific investigation and research methods is, to a large extent,
dependent upon the set of the "beliefs" that prevail in the community of researchers in any
given discipline [BAR 74]. Lakatos [LAK 70] argues that scientific progress is not about
the successive overthrow of dominant theories but a situation of "progressive and
degenerative problem shifts" and the outcome of rival research programmes in contention
(page 179). In HCI, the research methods used derive from significantly different
fundamental philosophical and scientific bases, from the proponents of predictive
modelling, e.g. [YOU 89], to those advocating prescriptive guidelines, tools and methods
[LON 89] and (more radically) the claim that "theories" are embedded in the products or
artifacts themselves [CAR 89] CAR 90] [ CAR 91]. However, many would argue that
these approaches, nevertheless, support both practice (in design and engineering) and the
theoretical underpinning vital to a strong scientific discipline [MAR 91].

A distinguishing feature of HCI research is the stress upon immediacy of results. The
demand for new products and effective design principles places an intense pressure upon
researchers to deliver immediately applicable knowledge that can be applied using the
existing technology. As a result, the need for well designed and engineered products often
takes precedence over the quest for long term scientific knowledge. The drive to improve
computer systems design by rapidly turning research results into products or artifacts has influenced the traditionalist approaches, as exemplified by the work of Card, Moran and Newell [CAR 83] and Norman [NOR 88], where the value of the methods advocated is couched in terms of applicability. It can be argued that the impetus to transfer HCI research results quickly into sound applied principles is in conflict with the longer term needs of basic science.

4.2.2.1 Counterviews to the Mainstream

The sources of inspiration for those researchers seeking alternatives to the conventional scientific paradigm are many and the philosophical and subject disciplines diverse, ranging from ethnomethodology and hermeneutics through general systems theory and socio-technical approaches and action research [GAR 67] [BER 68] [CHE 81] [CHE 90] [JAR 91 a&& b] [ARG 85] [WAR 80]. Suchman [SUC 87] and Winograd and Flores [WIN 86], for example, draw upon ethnomethodology, hermeneutics and conversation analysis for their theoretical frameworks.

Burrell and Morgan [BUR 79] differentiate between two extremes from the systematic techniques associated with formal mathematical analysis, laboratory and field experiments and survey techniques, to the action research and case study type, in which first hand knowledge is required about the subject under investigation. Another approach is to characterize case study, phenomenology and hermeneutics etc. as sensitive, exploratory approaches contrasted with theorem proving causal modelling, statistical tests and experiments [JAR 91a]. The polarities implied are of deterministic and non-deterministic approaches or rather from system focus to the human-centred.

Avison and Wood-Harper [AVI 91] describe the different approaches to computer system development. At the "people-oriented" end of the scale are the soft systems methodologies such as SSM [CHE 81], ETHICS [MUM 79] and Multi-View [AVI 90]. These approaches are concerned with delivering effective systems in different contexts and assume that there are a complex set of factors that must be addressed. At the opposite end of the scale, the data or "system-oriented" approaches and structured methods associated with traditional
systems analysis find less favour in the user-centred design methods more in currency in academic research establishments and a few large corporation human factors research groups. However, there is evidence that the human-centred approaches have not penetrated commercial practices yet because they do not fit with existing methods and practitioners do not perceive benefit [MCL 91].

The fundamental issue is one of the relationship between the generation of scientific theory and the applied context. There is a tension between achieving generalised principles or properties and the context dependency of the application. Thus, whilst Avison and Wood-Harper [AVI 91] argue convincingly that an participatory action research approach [LEW 51] WAR 80] is the best way to acquire knowledge that can be applied in context, they admit that it is subject to charges of bias and not likely to yield generalised results. The implication seems to be that you cannot have it both ways. Does this mean, in effect that participatory research offers no scope for the production of basic scientific knowledge? It can be argued that a reflexive approach [WOO 88] that recognizes a number of co-existing facets of scientific concern can offer a positive way forward. A key issue is the need for making participation of all the "stakeholders" an integral part of the research and design activities.

4.2.2.2 Participatory Research

The importance of participatory research is acknowledged in HCI and, indeed particularly in Scandinavia it has a long tradition [BJO 78] BJE 87]. In more recent times the close relationship between the research results and the transference into system design has been reflected in the terminology: for example, Contextual Enquiry becomes Contextual Design [WIX 90]. Participative design (see e.g. Muller [MUL 92] seeks ways of enabling customer or user driven design processes in order to ensure that their requirements are fully transferred into system design. The reasons for the introduction of user involvement in system design are many, as are the forms it takes [DAM 81] [MUM 79] MUM 81]. The trend towards user-centred design has been a consistent phenomenon of the 1980s [NOR 86].
In the main, the term 'participation' refers to the involvement of users in the design process and, usually in the early stages when the initial requirements are defined. There are well recognized problems arising from the fact that user views are not always articulated in a way that can be readily translated into system design characteristics. In addition, the initial requirements are unlikely to remain static, especially when the first prototype is available for evaluation. In order to maintain any acceptance or commitment gains arising from the early user involvement, it is important that the later changes are acted upon in the system design as it evolves. This need to ensure a flexible and responsive design and development activity continues to be a problem even with the advances in rapid prototyping tools largely because the organizational structures are implicated and these do not necessarily facilitate such a process.

Most of the interest in participation has concentrated upon the role of the prospective user of the system, either in the requirements activities and or the design and evaluation of the system itself. It is not always made explicit that other members of the team are equally implicated in the identification of the user requirements, the translation into design features and the transference of research results into design changes. In the experience drawn upon for this paper, the matter of participation extends to the system designer, the evaluator, the research investigator and any other person active in the project. All interested parties have participatory roles to play alongside the user and the explicit allocation of responsibility needs to be clear within the team.

If, as in the approach described below, the research agenda includes both gathering data about human-computer interaction and designing a system at the same time, there is a need to adopt a flexible research methodology. That methodology must be responsive to the emerging situation and yet provide tangible and immediate results that inform the particular case immediately. Deriving knowledge that is immediately applicable requires different techniques particularly if it has to address multiple issues in context which imply a different granularity of research problems to be addressed. The relevance to more general cases cannot be established through single case developments and, thus, does not in itself contribute to general scientific theory, except, perhaps as a counter example. However, for
the problem in hand, the importance of the immediate transfer of findings into the system design and the participation of all parties in the research activity must be recognized. In the following section, the research and design framework is described. It recognizes that a number of co-existing elements contribute in differing ways to the research process and that the roles of all the participants in the design team must be engaged throughout the iterative cycles of requirements definition, design, implementation, evaluation and re-design.

4.2.3 The Twin Paths of Research and Design

In participatory research, there is explicit recognition of the inter-related roles of the personnel involved, whether domain specialist /knowledge worker, the system designer/developer or the research investigator/evaluator. Within the broad framework set by common goals, these participants also have individual objectives and, to that end, may seek to apply different methods and techniques at any given point in the cycle of activities. The research and design activities operate as an iterative cycle of investigation, analysis, results and feedback into design and development.

In Knowledge Support System development, the overall goal is to design a knowledge system that can be used and extended by the domain specialist and also be evaluated in a manner that shapes the design of the system. In addition, the general process of such computer-based support is studied. Thus, reaching an initial point where the system can be utilised for real tasks rather than test cases is an essential prerequisite. From that point onwards, the study of its use "in anger" becomes a realistic possibility. At the same time, the subsequent studies will have implications for the design itself and such information needs to be gathered systematically and in such a form that is of use to the designers. The Research Cycle described below pre-supposes an earlier phase of requirements analysis, design and development of the KSS system.

Thus, the Research Cycle and Design Evolution described below are the twin paths of complex computer system design and use and the two are both reflexive and iterative in character. The first path focuses upon how the knowledge capture and extension by the
domain expert using a KSS takes place and what form it takes. The parallel path is that of the evolution of the KSS design in response to the domain expert's use of it and evaluation by other participants. This situation presupposes an initial requirements and design phase leading to the development of the first system as a basis for study. Whilst the twin activities are recognized to be co-reflexive, it is important also to note that the results lead to different outcomes (See Table 1).

4.2.3.1 Research Cycle

The research cycle consists of three types of activity: i) exploration, ii) observation iii) evaluation. The goal of the research path as distinct from the design path is to seek data from the full range of user tasks and, in analysis, to differentiate between those results and those that impinge directly upon design: so, for example, information about the way Visualization [EAR 92] is used to increase understanding and knowledge has more general relevance to say, the utility of the "mouse" for drawing lines upon the visual data. The three types of research described are not mutually exclusive: different goals and methods apply within each, however, and the participant roles vary according to the objectives.

Exploratory Studies

In the exploratory activities, the investigator, in close collaboration with the domain expert user or knowledge worker, seeks to identify the widest range of questions about the latter's tasks relating to knowledge capture and evolution both using the KSS and without it. It is most important that no significant assumptions or expectations are held that might inhibit the recognition of new significant issues. In this mode, we learn most about the primary issues of concern and the identification of the significant variables. The domain expert's activities are monitored using video and audio recordings and playback facilities employed for post hoc reflections on what took place. The first task of establishing which are the primary issues that warrant attention is the initial process of selection, or "construction" from the raw data collected. In any longitudinal study, because of the vast amounts of data available, there is good reason to select snapshots of activities that have been identified as reasonably typical of events.
The choice of research method for these purposes is a critical one. Participatory research is essential because it enables the inclusion of the domain specialist's views in a situation where external observation alone cannot access all the vital information. Exploratory studies of this kind do not assume existing hypotheses and may take account of the full range and complexity of the matters under investigation. However, being "data rich" is only the first step: it is then important to sift out the categories of information that are relevant to the main research interest and to the system design, and thence, to inform the next phase of research.

**Observational Studies**

Observational studies are made after an exploratory stage and employ structured methods to collect data about the issues previously identified. In this case, data can be collected in a number of ways but, typically, structured interviews of the knowledge worker by the investigator will form part of the study. In this mode, data is gathered about targeted issues: for example, user task goals and methods, usability of system interface, perception of the knowledge-base performance, results of domain investigations using the KSS etc. User interface design is particularly influenced by the outcomes of this work. Much of this work is employed to confirm or disconfirm the findings of the exploratory studies. Other participants in the team, such as an independent evaluator, may be brought in at this stage to offer a different perspective on the scope of the studies. The different perspectives are essentially subjective ones that may be tested only by a independent study of the primary source materials available on audio or video tape recordings. Interview transcripts also provide source material for different participants to scrutinize but it must be said that, whilst such raw data must be available, there may be a natural inclination to use results that have been produced by one person. This should be recognized but, nevertheless, resisted.

**Evaluation Studies**

Evaluation studies are performed by the domain specialist on the domain specific investigations and by the independent evaluator on the system design. These constitute the most formal studies in the process being described. The significant point is that the context in which these studies take place ensures that the strategy of testing hypotheses and
controlling variables can be applied to scientific advantage. At this stage, the variables and
the significant issues are understood enough for clear research questions to be posed in a
traditional experimental or empirical manner. In this mode, the knowledge worker learns
most about the domain questions whilst the research investigator examines detailed
questions about system usability and other specific design aspects of the system.

4.2.3.2 Design Evolution

The evolution of the system design is the second stream of activity. Here, the results of the
other processes are used to inform design and re-design of the system. Rather than
identifying issues of concern or answering questions that have emerged, the system
designers postulate and implement solutions based upon requirements gathering exercises
and the results of the observation and evaluation exercises from previous version of the
system. It is frequently necessary to re-enter the exploratory mode following advances in
this activity where the resulting changes in the system may be so extensive as to warrant a
fresh examination of the impact on the domain expert's knowledge work. Carried out
successfully, this form of continuous reassessment and evaluation can ensure well informed
design. The results are not demonstrated in theory but in the artifact of the system.

The design of the Knowledge Support System provides a case-study of a design activity as
well as a research exercise. The design of the system can be viewed as an artifact
incorporating implicit theoretical constructs that realize functional and operational
requirements [CAR 89]. Throughout the design process, features and dialogue structures
are chosen because of their ability to achieve the intended functionality; such decisions are
then evaluated against various task specific criteria. During the design process the
descriptions are modified and there is a clarification and refinement of intended functions
and requirements.

In the case in question, consultation with the users - the term "knowledge workers" is used
here [KID 94], took place in the early stages to check on the matching with task goals and
characteristics. The nature and goals of the knowledge worker's investigations were
unfolding as the experimental work took place and new requirements were emerging. The
resulting system was well received and proved to be productive in terms of its support to the current domain tasks [CAN 93].

The following phase was carried out with even greater user involvement and with more structured evaluation exercises. The new version had largely the same basic functionality with additional features: e.g. notepads, rule tracing, scrolling rule windows and additional graphical annotation markers. The user's degree of control over the knowledge base structure was identified as a fundamental question. The user interface had been completely re-designed but it was clear that despite increased user consultation and attention to evaluation there was still room for improvement. A different procedure for the new system was devised. More rigorous procedures than previously used were employed and methods for recording the development process in all areas of the project activities discussed. The participants' goals and methods within each research and design cycle are summarised in Table 1 below.

<table>
<thead>
<tr>
<th>RESEARCH CYCLE</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Exploratory Studies</strong></td>
</tr>
<tr>
<td>Participants: investigator, domain expert/ knowledge worker</td>
</tr>
<tr>
<td>Goals and Methods:</td>
</tr>
<tr>
<td>To identify questions about domain tasks</td>
</tr>
<tr>
<td>Do not assume existing hypotheses</td>
</tr>
<tr>
<td>Unstructured interviews, video/audio recordings</td>
</tr>
<tr>
<td>Snapshots of 'typical' activities</td>
</tr>
<tr>
<td>Include specialist's point of view</td>
</tr>
<tr>
<td>Selection, or 'construction' from the raw data</td>
</tr>
<tr>
<td>Sift for categories of information</td>
</tr>
<tr>
<td>Inform next phase of research</td>
</tr>
<tr>
<td><strong>Observational Studies</strong></td>
</tr>
<tr>
<td>Participants: investigator, domain expert/ knowledge worker</td>
</tr>
<tr>
<td>Goals and Methods:</td>
</tr>
<tr>
<td>To confirm or disconfirm the exploratory studies</td>
</tr>
</tbody>
</table>
Collect data about targeted issues e.g. system interface
Use structured techniques: different perspectives: scrutinize primary source materials on audio or video tape recordings, interview transcripts.

- **Evaluation Studies**
  Participants: independent evaluator, investigator, knowledge worker
  Goals and Methods:
  To pursue research questions about detailed issues in an experimental or empirical manner.
  Test hypotheses and run tests using controlled variables

**DESIGN EVOLUTION**
Participants: designer/developer, independent evaluator, investigator
Goals and Methods:
Results of the other activities inform design and re-design
Implements solutions from range of choices explored
Necessary to re-enter the exploratory mode following changes
Well informed design takes place.
Results produced in the artifact of the system

| Table 1: Participants' Goals and Methods |

4.2.3.3 **Participants' Outcomes**

In order to obtain research advances in the domain, in software design, in human-computer interaction and in scientific theory, a reflexive relationship between the research and design activities and the roles of the participants must be recognised. Whilst different results may well be presented to specific research communities as discrete elements (see references below), it is necessary, in the actual work, to be quite clear about the inter-relationships. The objectives and outcomes from the participants' work, are, nevertheless, quite distinct. The results of this process can be viewed from the different goals and scope of the participants in the process as summarised in Table 2 below.
For the knowledge workers, the expertise captured in the KSS in a rigorous machine tractable form is available for further study and for validation with other experts in the field [OBR 93]. At the same time, the development of a wider understanding of the domain and, in particular, the evolution of complex knowledge structures that can be explored and investigated further is significant result [OBR 92b].

For the system designers, what begins as an assignment in screen layout and graphical interaction methods evolves into a whole system re-evaluation and re-design and implementation [EDM 92a]. The design outcomes embodied in the KSS system are the result of applying new techniques for user interface and knowledge base construction [EDM 92b] EDM 93].

For the research investigators, the range of possible outcomes is wide. It may range from appropriate design features for the support systems to the structure and use of human
knowledge and the creative and innovative aspects therein. In particular, the relevance of these findings to general cases in other domains of expertise provides the basis for more general outcomes [EDM 92] CAN 92].

4.2.4 Conclusions

A case for participatory research that incorporates all the team participants' goals, methods and outcomes has been presented and an approach to Knowledge Support System (KSS) research and design proposed. It is suggested that a research cycle comprises three main stages within which distinct but inter-related methods are applied at different stages, from the open ended exploration of new ground to the close scrutiny of detailed issues using traditional experimental methods. This process is iterative and, where the evolving system design alters the context significantly, it may be necessary to return to earlier more exploratory approaches. The approach affords opportunity for a flexible response to change in the knowledge worker's requirements that may be fed into the system design.

A fundamental issue raised is that of the relationship between the generation of scientific theory and the applied context. There is a tension between achieving generalised principles or properties and the context dependency of the application. Participatory research action can derive results that are immediately applicable in the system design. The reflexive relationship between the system design and research into the user-system interaction and the context in which it takes place, may yield knowledge of a more general character. An outstanding question is whether the research cycle proposed has implications for the process of scientific research.

References


5.1 ISSUES IN THE DESIGN OF EXPERT SYSTEMS FOR BUSINESS


This paper is concerned with critical issues which have emerged from experience in the design and implementation of expert systems. The key elements of the design process which are vital to the success of the system are identified and discussed with reference to a recent collaborative project. It is suggested that, in the context of expert system applications, the conventional phased, incremental, approach has significant limitations. A design and development model is described which provides a framework that places user involvement and the establishment of system acceptability in a prime position. It also addresses the organisation of collaborative, team work in a complex and dynamic situation. It is recognised that there is no ideal generally applicable method for expert system development because the organisational context imposes its own imperatives within which priorities, resource allocation and deployment of personnel are decided.

Keywords: design process, life cycle models, user involvement, prototyping, evaluation, organisational context

5.1.1 The Design Process

This chapter is concerned with critical issues which have emerged from experience in the design and implementation of expert systems. The key elements of the design process which are vital to the success of the system are identified and discussed with reference to a recent collaborative project. Some current approaches to system design which arise from differing perspectives are surveyed briefly. It is suggested that, in the context of expert system applications, the conventional phased, incremental, approach has significant limitations. Amongst the complexity of the forces at work in a field development carried out by a number of collaborating groups. It is clear that the three most significant factors which affect the ultimate success or failure to meet project objectives are:
- the difficult nature of the requirements analysis process and its relationship to the knowledge acquisition activity.
- the requirements of different groups of users within the particular business environment and the way these impinge upon the system design and its evaluation.
- the impact that different perceptions of the design process within the development team can have on both the nature of the design concept and the way it is carried out.

These issues are thrown into sharp relief when the development team is faced with the problem of how to reconcile corporate aims, different departmental and user management objectives and end user needs. A design and development model is described which provides a framework that places user involvement and the establishment of system acceptability in a prime position. It also addresses the organisation of collaborative, team work in a complex and dynamic situation. In particular, a view of the role of prototyping and its relationship to system evaluation by users within the whole process is discussed. It is recognised that there is no ideal generally applicable method for expert system development because the organisational context imposes its own imperatives within which priorities, resource allocation and deployment of personnel are decided. An expert system development team which comes from outside that environment faces considerable demand on professional abilities ranging from communication and interpersonal skills to sheer ingenuity and resilience.

The Case Study: the ideas put forward in the discussion to follow arise from recent experiences in the development of an expert system for an estimating office within a UK manufacturing company. The context of the application was the estimation of machining costs prior to tender. The expert system was intended to be used by experienced cost estimators whose job it is to produce the estimates, usually to very tight timescales. In the existing manual system, the estimates appear as hand-written 'operation layouts': i.e. sets of machining actions required for a particular part (in this case, aircraft landing gear) which are then conveyed to management for further action. The application development group consisted of the lead partner, a software company, and two academic partners. This group was funded both to deliver a demonstrator application and to produce longer term research
results in the design and implementation of knowledge bases and their user interfaces. The overall project objective was to demonstrate the value of logic programming in the design of mainframe based expert systems integrated with conventional DP systems.

5.1.2 Design and Development Methods

By its nature, design is a complex activity in which the decisions that are made finally determine the characteristics of a product. The way in which successful products are achieved is understood only in a limited way. In the case of software products, the emphasis at this time is on the identification of explicit, often formal, strategies that can be employed. There is, however, considerable debate and disquiet about design strategies or, more accurately, an agreed standard strategy in the development of computer systems. A rapid expansion in applications, particularly of expert system technology, is currently taking place against a background of evidence that a majority of software systems developed commercially are "poorly specified, poorly designed, poorly documented, (and) difficult to use" [1].

Various design and development methods have evolved to meet the increasing complexity of applications in the business world. Some embody elements of traditional phased development, whilst others, such as structured analysis, stress the role of logical models that focus on what must be done rather than how it is done. More recently, the Structured Systems Analysis and Design Methodology (SSADM), developed as a standard for government applications, attempts to widen the scope of structured analysis and offers a prescription for a more rigorous conduct of system development [2]. For a useful summary of the design methodologies for constructing software systems from structured approaches such as top down functional decomposition, the data structure-oriented and object-oriented methods, stepwise refinement, graphical notations etc., see Wilkinson and Winterflood (op.cit.).

Depending on the application or research goal, the methods may focus on one or other point in the design and development life cycle. There are many divergent views as to efficacy and appropriateness. The legacy of traditional systems analysis, based as it is on
developing successively tighter definitions of the proposed application until the final series of programs is obtained, affects the way people approach current applications. In the minds of many, the analysis and design are separate and consecutive processes:

"Analysis specifies what the system should do. Design states how to accomplish the objectives" [3].

But, the concept of what design is, which underpins the basic philosophy of the arguments put forward here, challenges that definition and draws from the discipline of design in the broader sense. Design is, in reality, a co-ordinating activity which encompasses task analysis, requirements definition and implementation. It is particularly important to understand in relation to expert systems that requirements definition cannot be dealt with as a discrete stage prior to development (see 4.1).

In this century, there has been a considerable history of ideas relevant to the understanding and development of systems. It has not always been clear that computer science has fully taken account of that history. Checkland [4] reviews the background of the development of systems thinking in the 1950s and 1960s to the attempt to apply these ideas to complex human activity systems in the late 1970s and the subsequent soft systems methodology which he characterises as "a formal way of moving from finding out about a real-world situation, which some actors regard as problematical, to taking action in the situation.". The approach assumes the need for continuous debate where the conceptual model is seen not as a model of the situation but one relevant to it and to be used as a means of comparing alternative perceptions with a view to change. He is, of course, referring to information systems development but the recognition of the existence of uncertainty and the dynamic environment applies even more to expert system development where the territory is entirely new and where the end users are often professional and expert.

Approaches, such as Multi-View for information systems definition, attempt to encompass the human dimensions of system design [5]. The Multi-View method is distinguished by the attempt to take account of the different points of view of everyone involved in analysis and design of information systems, by combining an analysis of human activity, socio-technical systems, data analysis and structural analysis. Similarly, Shackel
[6] puts forward the ergonomic approach to design as an iterative process involving systems analysis, workstation (or task) analysis and evaluation. Usability is defined in terms of the effectiveness with which a system allows a task to be carried out, its learnability, flexibility and the attitudes it engenders in users. Design in this sense, involves both human factors and software engineering. The notion of usability must be defined in the context of the total design methodology: the criteria for usability are determined by the specific user and task requirements. It must also satisfy the objectives of the corporate body, user management and, perhaps, the computing department of the business.

It is apparent from the range and diversity of these methodologies that there is no one agreed way. It could be argued that attempts to find one best way to deal with a wide variety of applications are misguided and lead to "elaborate and bureaucratic methodologies" [7]. The authors propose a solution based upon toolkits. However, solutions which centre on particular toolkits are inadequate [8]. There is a need to provide a flexible framework within which a variety of tools can be applied where and when appropriate. In addition, it is clear that system design must take account of the impact of the system on the organisation and, in a sense, becomes an aspect of the management of change. The guiding design principles must be clear and carefully implemented in order that the coherent philosophy is understood and carried through by all participants [9]. This is easier said than done and the tendency to adopt reductionist approaches for applications reflects the difficulty of addressing all aspects, from technical and resource considerations to the broad socio-political issues such as organisational and user requirements. The problems are accentuated in the expert system field where the prospective users are more often people with specialised, professional skills. However, where specific methods have been devised, the focus has been on the problems of knowledge acquisition and the structuring and representation of the knowledge acquired: the concomitant issues of user requirements and design for usability and acceptability have received considerably less attention.

Experience shows that the exclusion of the users from the design of a system has serious implications for loss of time due to lack of acceptance and resistance to change. The lack of
consideration of the end users in particular has played a part in the poor performance of some IT systems developments to date [10]. It is all too easy to overlook the complexity of user requirements. In particular, the professional practitioner's requirements are not easily defined and, in relation to the impact of support systems are ever-changing [11,12]. Edmonds [13] argues that the notion of a "total design" concept contributes to understanding the extent of that complexity. Not only is it impossible for the designer to master all the specialised skills of the system builder, it is also impossible to master all the specialisations from which factors contribute to design decisions. The designer must coordinate a variety of specialists in a process of synthetic production, and integrate their work into its wider context.

The integration aspect and the introduction of such a system into a live business environment make the issue of how to bring about user involvement a more difficult matter. The wide variation in users and the evident complexity of addressing their requirements deter many developers from tackling the problem. There are a number of different ways in which users can be brought into the design process or excluded from it as the case may be [14]. User representatives, for example, provide controllable user expertise and are, inevitably, influenced by the design team. Participative design where users are involved in selecting the form of work organisation but not the technical design or programming is known to be successful in engaging users' commitment [15]. However, user-centred design, in which users design and experts advise, is a practice which is rarely seen. The critical issue is not who does the work but who has the power to influence the design. All sections of users need to be represented and this is particularly important where there might be more than one user group with varying requirements of the system. The main problem is that there is no reason to suppose that corporate management, user department management, data processing department and groups of end users, will agree on where to strike this balance. Where the end users are experts or professional people in their own right, it may be that the only way to design an acceptable system is to centre the process on them, their task scope and operational conditions. However, it must be remembered that the product of a user-driven development can be quite different from that of a knowledge based system/expert system driven process.
One might note that the development of methods in this area has come from general work in information systems rather than from the expert systems field itself. However, expert systems pose particular problems that may not be fully catered for in the more general methods. These problems relate to the much stronger emphasis on flexible systems for professional users that we have mentioned above when discussing the work of Checkland and others. It seems that we may well be able to draw upon research in general systems theory, but such actions will imply different approaches to new and changing human-system environments. Most of what follows should be viewed in the light of these comments.

5.1.3 Expert Systems Design

In expert system development, much attention has been given to the problems of knowledge acquisition and representation. The effort is directed towards identifying and interpreting expert knowledge from interviews with experts (or textbooks) followed by mapping directly from verbal data to a chosen form of representation (e.g. rules). Knowledge acquisition is carried out at the very beginning because of the need to understand how to structure the knowledge and the inferencing mechanism. The nature of the knowledge required is important to define. In this complex process, there is much scope for misrepresentation [16]. More seriously, attention to these difficult areas usually takes precedence over how and by whom the system is to be used. In the KADS project which aims to tackle knowledge based system development methodology on a broad front, recognising that "partial advice is worse than useless to the system developer", it is acknowledged that their analysis techniques concentrate on the modelling of expertise, and that the modelling of user-system interaction requirements is left ill-defined [17]. Since that paper was written, attention has been given to modelling in other areas, notably what, in KADS, is known as architectural design and modality analysis. The latter addresses the issue of how the theoretical work is to be incorporated into KADS and validated.

There are factors such as the initial focus of attention and overall goals which pre-determine the knowledge engineer's view and, thereby, or the system design concept itself. In the case study in point, the knowledge acquisition activity concentrated on determining and
categorising knowledge manifest in the final stage of the user's task process, i.e. the writing out of the operation layout [19]. The resulting analysis led to both an expansion of that component and a significant change in its implementation. When requirements analysis for the user interface was carried out later, doubts about the concentration of system support for this aspect arose. This was because that requirements analysis included consideration of the task operational features, such as the relative time spent on different stages in the estimating process as well as the information needed at various times.

An important consideration was that the system in the example under discussion was intended to be a co-operative one. Co-operative expert systems aim to provide assistance or support in a way which broadly conforms to a user's existing methods and normally allows the expert to make decisions during the consultation process which may affect the advice given subsequently. The user's methods in carrying out a given task must be taken account of in the overall design of the operational aspects of the user-system relationship. Thus, the process of co-operation is significant in itself, as well as the functions made available and the accuracy of the advice given. In the present state of computer systems, it is probably more appropriate to talk about human co-operation and system response. The ideal of the co-operating computer is still to be attained.

Nevertheless, it must not be assumed that a user-centred design approach will result in a co-operative expert system. It is sometimes more attractive to users to operate in a simple well defined manner that could not be called co-operative in the strict sense. However, both approaches represent a shift away from a fixation on the autonomous expert in expert system research, towards a greater emphasis on user involvement. In the case of co-operative problem solving systems, this relates to the involvement of users in the reasoning processes of expert systems [20]. But equally important in practice is the way the expert knowledge appears to the user. If the representation of the knowledge is carried out separately and prior to the total system design, there may well be irreconcilable conflict. Where the two are in conflict our experience suggests that the user-centred design approach should take precedence. In user-centred design, the designer would need to advise the users.
about the available co-operative problem solving options but not impose particular solutions based on a functional analysis of the situation.

Thus, a co-operative problem solving approach is only likely to facilitate user acceptance if:

1. it is closely modelled on how users and experts actually co-operate in a particular task domain and
2. the users actually want a computer system that models this form of co-operative assistance.

This last point is of special relevance to expert users who are likely to be very reluctant to see the interesting elements of their work subsumed within a system but who, by contrast, may welcome the provision of simple on-line information support. On the other hand, user management may be keen to see some of the expertise available in this form because it could provide them with a degree of standardisation and could also be useful in training new recruits. Thus it is clear that there may be conflicts that expert system designers must resolve.

5.1.4 Key Elements of the Design Process

The attention paid to the different aspects of system development varies according to the circumstances and the objectives of the power agents in any given project. Very often a key individual with enthusiasm and drive is to be found behind the successful scheme. If a system is to become more than someone's pet project and perform a productive commercial function, it goes without saying that it should be used. All too often, the importance of end users in the measurement of success or otherwise is notional. To ensure sufficient attention to user requirements analysis and specification and to place it correctly in relation to other activities, such as knowledge acquisition, it must be included explicitly in the project plan. It is not an easy task to ensure that users are taken account of, let alone be directly involved, in design. However, some activities lend themselves more readily than others to enabling the user voice to be heard. The approach to evaluation and prototyping as vehicles for increasing user involvement and helping re-design the system
is an important part of any satisfactory strategy [21]. In combination, user evaluation and the deployment of the user interface prototype as a vehicle for clarifying user requirements, provide practical mechanisms for assessing the appropriateness of the whole system design. A deeper exploration of the operational and functional aspects of the proposed system in relation to the total user task scope must then follow. In this section, we consider certain key elements in the process.

5.1.4.1 Requirements Analysis
The specification of user requirements is a difficult problem. Requirements that arise from the early task analysis work may well contain errors or misconceptions on the part of the developers. A failure to apprehend the degree of complexity of user requirements is often the root of the problem [13]. It is doubtful whether methodologies which are mainly directed at the computer core of the application, such as SSADM, address the area of requirements analysis adequately. Indeed, there is evidence that the most successful systems analysts pay attention to the socio-political aspects [22].

It is also worth bearing in mind that the word 'requirement' implies uncovering something which is already there in the minds of prospective users and management. This assumption can be mistaken on two counts: firstly, where a 'requirement' is discovered, it may not be very clearly articulated or, indeed, may be expressed in vague terms (e.g. 'the output must exhibit 5-10% accuracy) because that is the current figure for human performance' stated without the provision of a method for measuring accuracy; secondly, and commonly in the case of people with no previous experience of computer use, there is a difficulty in imagining what support a computer system could offer, particularly in the case of highly complex work, in other words, there is in fact no requirement.

Another important factor is that requirements are constantly changing: by the very act of introducing a computer system into the thinking of the users, this creates an impetus to perceive tasks and relationships in a more dynamic manner. This will be increased with the first and ensuing contact with any early versions of the system. Chasing the changes in
user requirements is traditionally one of the nightmares of system designers. How much of the well recognised high cost of maintenance arises from this factor?

There are a number of misconceptions about identifying user requirements, in particular, the notion that users can visualise the proposed system from documentation or a few screen designs. Another misconception, that a user can evaluate a system without a user interface, is evident in the guidelines on expert systems for the Civil Service. The development phases identified reflect the traditional system development approach and preclude early user evaluation of prototypes by specifically excluding the user interface until after phase two [23].

Design strategies which take full account of the complete business environment are needed [24]. In particular, there is a need to see requirements specification involving prospective end users as an integral part of the design process. This is particularly important in relation to the early stages. The key point is that end user considerations in those early stages are vital to getting the initial design concept right, as distinct from the nature of the domain knowledge or details of the system functionality. Another important issue to be determined at that stage is where boundaries of the human and machine are drawn. It is necessary to determine the place of the system within the total task and user scope which includes understanding how manual and computerised activities relate.

The complexities involved in the provision of co-operative expert systems to professional workers are such that the requirements specification can only be completed after the users have experienced prototype systems. Indeed, considerable development progress may be needed before requirements can be completely finalised.

5.1.4.2 The User Interface as a Requirements Elicitation Tool

As we have seen above, when a new computer system is to be designed, it is often difficult to gather the requirements accurately. In modern computer systems the user interface is often a separable module that can actually be run, in some sense, on its own. The user interface goes deeper into the system than the screen and keyboard [25,26]. This
fact can offer an opportunity to assist with the problem discussed above. It is possible to prototype the user interface module and to allow users to operate it. Discussions with them concerning their actual requirements around this prototype may yield quite different results from studies based purely upon a paper exercise. Such a prototype can represent most of the operational aspects of a system, even though its functional responses may be limited, by example only or not available. The evaluator can, without major problems, explain such limitations to the user.

Hekmatpour and Ince [27] describe a project where a working version of a system was provided throughout its development for user evaluation, training or a starting point for developmental activities. This is an example of evolutionary prototyping rather than a phase oriented paradigm. During each iteration, re-specification, re-design, re-implementation and re-evaluation take place. In the case study under discussion, the intention was to treat the first prototype as a means of clarifying requirements and then discard it. When an acceptable design is achieved, the team then moves into the evolutionary development stage. Detailed functional and architectural design, and specification of the presentation aspects and operational behaviour of the user interface, form part of a rigorous software engineering exercise where the emphasis is on design for modifiability. Knowledge acquisition continues, within the structures elaborated and proven in the exploratory phase, and now embedded in the engineered system architecture. The first version should be sufficiently advanced to put into operational use: in the traditional life cycle this is the end of development and the beginning of maintenance. In the exploratory design cycle described in 6, this is the first evaluation and design review point in the process: changes, tuning, etc. will be followed by a series of such iterations until the evaluation role diminishes into a watching brief.

In the application of expert systems in general, it has been the knowledge base that has been the subject of the prototyping emphasis, in order to refine its accuracy. There is an earlier need for prototyping in projects such as this one - using the user interface as the prototyping focus, in order to elicit requirements from users. Prototyping in this way is, of course, distinct in kind from prototyping to refine the design. It is, therefore, important
that there is a common understanding of what approach to prototyping is to be used within the development team and that this is conveyed to the users from the outset. Initial perceptions of the system of an early limited function version of the final system can lead to confusion about what to expect and will prove to be difficult to change in later stages. Where there are high expectations, any signs of poor performance of the prototype leads to a lack of confidence in the development.

User interface requirements definition typically takes place within the design context established by the knowledge base analysis and, because of this, the end-users may be asked to define their requirements basically in terms of screen layouts. The user interface design is seen as a front end to functions already determined. However, users can have a limited awareness of the intended functions [28]. There is a common view in computing that the user interface component is the 'icing on the cake' and this view of the interface, the surface level of the design concept, was clearly shown to be untenable by the evaluation exercise. It had become apparent that the original task analysis and knowledge elicitation, oriented as they were towards the domain knowledge rather than the users' task processes, and management views before those of end-users, had resulted in a proposed design which did not provide the kind of support to estimating originally envisaged.

Several reasons are given by Ince and Hekmatpour for the necessity of prototyping the user interface design [29]:

- user interface formal specification can be very difficult (e.g. a written document does not enable users to visualise the system in use).
- there is a variation in the types and styles of users for any system.
- the complexity of requirements often leads to conflicting design goals which cannot be detected or resolved by written documentation.
- properties such as user-friendliness and ease of use are highly subjective, the 'look and feel' is revealed only when the system is 'live'.

The prototyping approach deals with these difficulties by allowing the design to iterate and involve end-users. In the early stages, it is argued that the emphasis of the
prototyping should be on the user interface. The reasons for prototyping the user interface relate directly to the dynamic and multiple nature of user requirements. Firstly, it is argued that the establishment of the users' operational requirements can be best explored and evaluated through the prototyping of the user interface both at the beginning of requirements analysis and throughout the design process. In addition, it is important to recognise that operational requirements can differ from the functional requirements. The latter are usually comparatively easily obtained, but it may only be by allowing end-users to experiment with the system that they can articulate the operational requirements.

Secondly, if this kind of development process is to be used, the prototype user interface must be deliverable in the field, and it must be a reasonable view of the proposed system in order to allow the end users to evaluate it in the light of their experiences. It is an important consideration at this point to remember that the requirements may belong to different groups, and that these requirements may be very different, or even opposed in their natures [30]. This may be due to different perceptions of what a particular system is for. Requirements initially agreed with the management, and which include new disciplines such as standardised terminology are likely to make the system restrictive from the point of view of the end-user. Installation of the prototype user interface will enable sufficient understanding of the proposed system for such issues to become explicit, be challenged and, it is hoped, resolved.

Since requirements analysis in a complex task domain can clearly be thought of as a particular example of knowledge elicitation, it is interesting to note the experiments, for example, by Berry and Broadbent [31], which show that verbalisation may not accurately reflect the most significant aspects of a task. There are, in any case, problems associated with knowledge elicitation which cause difficulties:

- experts may not tell the elicitor all the relevant details.
- they may assume that certain facts are obvious.
- their subject of expertise may be too complex for them to tell the implementors all the details except by asserting or refuting specific examples.
Given that verbalisation alone can be misleading or incomplete, the user interface can provide a concrete object to facilitate both discussion and observation relevant to the knowledge that stems from requirements analysis. It can also provide a means of refining this knowledge, and, in a realistic way, give end users something to complain about! In this way, general requirements can be refined into specific ones in a series of iterations. While a good user interface is desirable, the real value in using the user interface in this manner is to enable the end-users to talk about what they want the system to do, and to detect problems before the main system is built. Prototyping to define requirements in this way is not a new idea, but from the experience of this project, prototyping the user interface alone will facilitate a majority of the requirements definition. A similar view is taken from the Software Engineering point in the USE methodology [32]. If the knowledge base is built first, the user interface must simply reflect its functionality, and can therefore only be designed at a presentation level, which is well understood to be an incomplete view.

5.1.4.3 Knowledge Acquisition

Knowledge based systems methodologies [17] have tended to focus on knowledge acquisition as the major development task. The objective being to construct a conceptual model of the expertise in some task domain. Requirements analysis, including establishing the appropriate role of the system, have tended to be regarded as separate and secondary activities. This is precisely the contrast that we have discussed above in relation to approaches to co-operative systems. An alternative design strategy is one where knowledge acquisition is directed to obtaining just that knowledge required to fulfil a system's intended role. Such an approach was adopted in the prototype of the case study, where an early decision was made to develop a co-operative problem solving system - supplying prompts, suggestions and explanations as required, and critiquing user inputs. The subsequent knowledge acquisition activity was targeted at supporting these forms of assistance. This is reflected in the acquisition techniques employed outlined below:

Informal interviews helped establish an initial map of the domain, including the main domain entities (machines, operations, parts and material) and the relations between
them. In particular, this technique provided information about the main determinants of operation layout such as material type and form, part type and configuration machine technology.

An analysis of estimate forms enabled the examination of the operation layouts for complex components which deepened the understanding about the relationship of the estimate/part/materials parameters to certain operational sequences. These provided a basis which could be verified in interview sessions with experts.

Talking through estimates provided an informal version of protocol analysis. While completing an operation layout from an engineering drawing the estimator was asked to report on his decision processes. The knowledge engineers took notes and intervened for clarification. In retrospect, this technique may have been overused - the quality and quantity of knowledge elicited was limited given its time-consuming nature.

Standard forms, such as matrices drawn up to capture all possible permutations of: machines against actions, stage of manufacture against actions, and so forth were completed. This knowledge was used in the generation of the dynamic action menus, and in the interactive checking of operations.

However, as indicated previously, these techniques on their own were insufficient to ensure that the prospective end user requirements were established in the early stages of the design process. In particular, the relative importance placed upon different stages in the estimating process and the importance of information derived directly from drawings within that, was not fully understood.

The knowledge needed to understand the requirements of the users performing the task is much broader than the knowledge required to adequately understand the functional answers that must be provided to the problems posed. This point is, of-course, particularly significant in the case of a co-operative expert system. A concern for requirements must clearly figure highly in the early stages of a project. But it must be
borne in mind that this is only the start of ensuring the design is satisfactory. The emphasis on "getting the requirements right" from the outset is misguided [33] and, by implication, the knowledge acquisition process is more complex than is sometimes proposed.

5.1.5 Evaluation: the Case Study

The evaluation exercise within a user-centred design process is important in two particular respects. Firstly, it is an aid to arriving at the most satisfactory design in a basic sense and, hence, getting it right in a complete sense; and secondly, it is a means to facilitating user involvement. The two elements are, of course, inter-related. It cannot be assumed that even where experts are brought into the requirements analysis and knowledge elicitation work that the resulting design will be appropriate. Early user involvement in the evaluation of the first prototype and subsequent versions is essential. One element, which can be carried out by an agent outside the development team, is to introduce a quality assurance initiative as a means of early constructive support rather than only providing detailed criticism at the end. This kind of intervention can also be used throughout the design and development process to ensure that requirements are implemented and that vital matters such as the setting of acceptability criteria are attended to.

The scope of the evaluation is, ideally, extended to the whole environment into which the system is being introduced and the complete task attributes of the users, rather than being confined to the user-system interaction [34]. In establishing the general aims of the evaluation exercise, it is necessary to define them in terms of suitability for the task on hand from the end user perspective. The first goal of user evaluation is to provide feedback on the design and operation of the system and convey the results into the next stage of the design and implementation work. The scope of what is to be evaluated will depend upon the context but will usually include general design features of the user interface, the accuracy and appropriateness of the expert system advice, the reliability, performance, acceptability and ease of use of whole system. In establishing evaluation criteria with users and user management, it is important that appropriate assistance is
provided by an evaluation co-ordinator whose role is to support and record rather than to impose prescriptions or ready made, inflexible guidelines. In addition, the development team will, of necessity, need to establish its own evaluation criteria and to relate these to the users' own.

Collecting clear accurate data which can be readily transferred to the next stage of the design is very important. Structured interview formats devised from taped unstructured interviews can provide more accessible results than ad hoc conversations. However, feedback during initial training or informal discussion about the early system use and related activities is useful to monitor. The users, in the case in point, were encouraged to record their observations about the prototype user interface on a 'scratch pad' or write notes during use. In addition, automatic time stamped recording took place. Using the data gathered from these different sources provided a more complete picture of events. Because of the possibility of variation in views between management and end users, the evaluation co-ordinator(s) must be alert to conflicts of opinion.

The evaluation of the stand alone user interface must, primarily involve end users. Their reaction, faced with the first direct experience of the projected system, can confirm any doubts about the matching of the design to their needs and, be a means of articulating requirements. The possibility that the users might not use the system, if it is the case, is then apparent to the whole team. A complete reassessment of the design concept is only likely to be made when it is apparent that the users are likely to reject the system.

The installation of a stand alone user interface in this case study made it possible to evaluate that component separately from the knowledge base. One advantage was the early identification of issues requiring change in the user interface. It was, thus, possible to carry out a limited number of changes prior to the installation of the whole system. With the linking of the user interface and knowledge base, the identification of those design functions and features attributable to the expert system and the architecture as distinct from the user interface was possible. In a later stage, when the expert system output was evaluated independently of system use, inspite of the high degree of
accuracy, users were not readily convinced that they could use it. Thus, it appears, that an evaluation of functional performance, in the form of accuracy of advice given, without direct operational experience with the whole system has limited value from an end user perspective.

Separate user interface evaluation also provided an opportunity for the users to become familiar with the software and for the development team to gain immediate feedback about its use and appropriateness. However, that there would be differences when the full system was installed was made clear. As a result of the user interface evaluation exercise a number of changes were made prior to complete installation. The changes made were of limited impact on the design of the user interface. They ranged from alterations to wording and type of input required to modifications in the specification of machines from individual machines to machine groups.

Some issues were found to be of paramount importance to users. In particular, the reliability and performance of the prototype system which was perceived by some as Mark I of an evolving system inspite of protestations to the contrary. It is essential that, even with a prototype, it must be reliable because of the likely impact on user cooperation and confidence. A lack of robustness and reliability in the system can influence the results of the evaluation in two ways: firstly, by preventing the users from carrying out a sufficiently thorough evaluation and, secondly, by affecting the users' attitudes to the system as a whole and their confidence in the development. The performance of the expert system, in the sense of its level of expertise, the accuracy and completeness of its advice, was critically important to user management. The prospective end users were more concerned about response times and the length of time taken to complete the whole task using the system. More important than this was the information such an evaluation yielded in relation to the design concept itself. This led to a reconsideration of aspects of the whole system and of the design and development process being employed.

Issues which emerged more explicitly included the difference of perspective between the end users and management. The latter had a more positive view of the system design and
thought that more training would overcome initial resistance. On the other hand, the prospective end users did not see how they could usefully incorporate it into the estimating task because it focused on the final and, from their perspective, least significant part of the process. The general view was that the method used was likely to slow them down. It was clear that the performance of the system was critical in a number of respects, not least of which was the accuracy and completeness of the advice from the expert system, which in itself, can reduce the need for user actions.

It was the evaluation of the user interface that was instrumental in making explicit issues concerning the design of the whole system, as described above, and it became clear that the earlier process used had allowed some significant matters to remain implicit for too long. The immediate outcome was a change in project organisation and procedures. The new design process adopted places more emphasis on end users and the user interface early in the process. The application development teams were made inter-establishment, divided into design and technical rather than user interface and knowledge base and the co-ordinating roles included a user company representative.

5.1.6 The Design and Development Model

From the experiences in expert system development that have given rise to the views expressed above, it can be seen that there are a number of overlapping elements which must be reconciled. In order to address the problems associated with requirements analysis and knowledge acquisition in the context of user centred design and organisational realities, it is necessary to adopt a development model which recognises the complexity of that process. The phased incremental approach is limited because it supposes that the initial design concept is basically correct and that subsequent iterations will involve amendments to the first implementation. The model adopted as a result of the project described supports two main discrete stages: firstly, an exploratory design prototype which is thrown away and secondly, an evolutionary prototype arising from the ashes of the initial design, which is incrementally improved. It is, in essence, consistent with that of [35]. See Figure 1.
In the feasibility study, the application is identified [24], the design team established, the corporate context determined, and the feasibility of a project assessed. Assessment of feasibility involves establishing the business case, ensuring that the system can deliver significant perceived benefit to the company, planning the project and organising the design team. Having determined the business case, the design team then carries out the exploratory user task analysis and the first definition of requirements. Once the initial round of knowledge engineering and architectures work is complete, the first prototype is constructed quickly, using the most appropriate tools for the purpose. These do not have to conform to the final delivery hardware or software base. Once the prototype has been evaluated, it is discarded. When the team is confident of the design, the approach becomes an evolutionary one whereby each successive design is evaluated and refined until it is deemed to be satisfactory.
A critical difference in this model from the initial approach used in the case study, is that the exploratory user requirements specification predominantly precedes the knowledge acquisition activity. From the task analysis and requirements exploration, a more detailed analysis of the task scope and user requirements emerged than had previously been obtained. Thereafter, additional knowledge acquisition is warranted, there being a much clearer picture of what is required in terms of procedural, operational and factual knowledge: for example, it was discovered that more knowledge about the role of drawing information in specifying the set of machining actions on a part, (an operation layout) had to be elicited. In this way, the user requirements analysis and specification impinges directly on knowledge acquisition work. The requirements specification defines the information, transformations and communications involved in the task, requiring knowledge based support and conventional information access, and, hence, delineates for the knowledge engineer, the scope of the knowledge required. In this case, the software architecture and specific knowledge representation employed in the first prototype were discarded whilst the domain knowledge itself was carried forward to be refined and extended in the evolutionary prototype phase.

The establishment of ground rules that can be agreed no matter which culture the participants come from is vital in the early days of the project. For example:

- involve the end users as early, as often and as much as possible.
- iterate round an elicit/design/prototype/evaluate cycle in design and development.
- educate, inform, and manage the expectations of management at all levels.
- agree a coherent design and development methodology and organisation and apply it.

The design team must include end user representatives, responsible for ensuring that a useful, usable system is produced, an expert "owner" of the knowledge and experienced systems professionals. The corporate context in which the system will be deployed is important and should be included in the task analysis and evaluation. The project's objectives must be integrated with corporate strategies and goals, and the criteria by which its success will be judged must be defined. Both corporate and line management should be involved in this process, and should be asked to make clear the scope that the
project team has for engineering changes in reporting structures, working practices and organisation.

Plans must be kept realistic and achievable. Some useful guidelines are:

- the horizon for detailed planning should not be more than about three months.
- overall project objectives should be stated in fairly general terms (rather than specifying detailed requirements at the outset).
- keep the use of milestones and specific deliverables limited to the detailed planning horizon.
- overall project objectives should be formally reviewed in line with the planning cycle.

In applying the model, it is necessary that the requirements and contributions of corporate management, line management, domain experts and computer department management be recognised and reconciled with those of the end users. But with such a broad spectrum of contributors to the design, it is inevitable that requirements will change during the life of the project. Changes can be caused by staff turnover, real-world changes, education, and the on-going development of ideas brought about by interaction with prototypes, introduction of early versions of the system, or just passing time. The oldest chestnut of software engineering - that "designing to meet a requirement is like walking on water - it helps if it is frozen" was always an impractical ideal: when developing systems with knowledge-based components, it is essential to adopt a design and development model which recognises and accommodates requirement and design changes as an inherent part of the development process. It is also important to recognise that the design and development model itself is only a framework which will be adapted to the particular circumstances of a project. Different companies have different procedures for defining and funding projects: departmental managers have different degrees of freedom as to how far they can take projects without corporate backing. Company management styles vary from the democratic and problem-oriented to the bureaucratic and dogmatic. Thus, for example, giving its end users an absolute right of veto (or even a say) over a system design may seem obvious and natural in one company, and be regarded as runaway anarchy in another.
5.1.7 Conclusion

In a case study concerned with the introduction of an expert system into a manufacturing environment certain important factors for success have been identified or emphasised.

It is particularly important that the design and development process focus on certain problems:

- the requirements of different classes of users
- the relationship of knowledge acquisition to requirements analysis
- mechanisms for enabling users to articulate requirements
- the organisational context.

A design model has been presented in which the design team (which must include user representatives) an iterative process. In this model, the identification and refinement of requirements is seen to continue well into the system development process. A particular feature that can be employed is the use of a prototype stand alone user interface to assist with the early requirements analysis.

Underlying the work reported here has been a concern to learn from Systems Theory and it is suggested that Checkland's notion [4] of soft systems methodology needs urgent attention in the context of the debate about design and development methods for expert systems in particular and computer systems in general.

Acknowledgements

The authors are members of a much larger team concerned with all aspects of the project. They are grateful to all members of the team, especially the users at Dowty Rotol plc. Particular mention must be made of Mark Buckley who has made valuable contributions to the case study of the project and commented on earlier drafts of this chapter. The work was funded as part of the Alvey Programme as project Alvey/IKBS/058. A version of this paper was published in Expert Systems: Human Issues edited by Anna Hart and Dianne Berry and published by Chapman and Hall: London, 1990.
References


5.2 A STRATEGY AND TECHNOLOGY FOR FRONT END SYSTEM DEVELOPMENT


This paper describes an approach to the enhancement of existing software and the development of new applications based upon the premise that advanced software technology is not in itself sufficient to realise high quality usable systems. Development strategies designed to ensure quality must be accompanied by appropriate system architectures and effective implementation tools. A technology and strategy is described that together enable the efficient development of user and task support systems in a wide variety of contexts. The approach and outcomes were developed in a major European project in which the needs of the user companies drove the direction of the outcomes which have since been applied in another industrial context (DEC).

Keywords: user centred design, development strategy, evaluation methods

5.2.1 Introduction

There exist a wide variety of situations where the complexity of the users' activities and the range of support applications are such that explicit task support must be integrated within the total system. For example, in the field of scientific computation, it has been recognised for some time that the needs of users are not being met by most current software applications [1]. For reasons of economy and the poor availability of such expertise, it is necessary to bridge the gulf between user needs and system solutions and yet to make use of the vast amount of reliable and comprehensive software packages already on the market. Advanced technology such as graphical user interfaces and multi-media may enable developers to construct better users interfaces to existing systems. However, we argue that this must go further than providing mere "facades" [2]. If the system is to be tailored to the
needs and task of professional expert users, there is a need for task specific support which is achieved by mapping the users' expertise and the functionality of existing applications.

The work described is concerned with the development of complex systems that employ reusability, task support, and the integration of application functionality. An important characteristic of the target users is that they are skilled professionals employed in critical problem-solving roles and, as such they have considerable discretion as to how they complete their tasks. Therefore, they are at liberty to refuse to use a software system that is not suitably tailored to their needs. In consequence they need to be closely involved in a user-centred strategy for system development.

The technology discussed in this paper is the Front End System technology (FES) [3]. The FES architecture and tools originated in Interactive Systems development, including and especially, User Interface Management (UIM) systems [4]. The development strategy and associated methods are drawn from a wide spectrum of studies in human-computer interaction and practical applications in the general software engineering field. The strategy and technology were developed in tandem and tested in industrial contexts as part of a large ESPRIT2 project.

5.2.2 Strategy and Methods

This section summarises the overall life cycle strategy for FES development. The key concept that underpins the strategy is iterative design within which the role of prototyping and evaluation during the process are key elements as shown in Figure 1 below. The approach is one of user-centred design drawing on existing expertise [5] and a concern to offer cost-effective and practical methods for industrial contexts [6].
5.2.2.1 *Iterative Development Strategy*

A significant amount of effort must be invested in the analysis of user needs and requirements early prior to any design or implementation work [7]. While the information captured in analysis will be invaluable in supporting the system design process, it will not necessarily be sufficient to enable the design team to "get it right first time". It is often the case that information about the users' requirements will not have been captured fully during the analysis activities. That information will be obtainable only after a prototype is available to evaluate with users. With conventional approaches to development, the effort that is invested in designing and implementing a system prototype is such that it is difficult to make substantial changes and remain within the budget for the project. It is necessary, therefore, to have an early phase of rapid growth of learning that combines analysis and design with evaluations. A second phase where a number of key decisions are made takes place before further development and evaluation activity. During this phase, it is still possible to make changes but they become increasingly superficial as the knowledge is refined.

5.2.2.2 *Prototyping*

One of the critical factors in making iterative development a practical possibility is whether or not one can generate prototypes or simulations that embody minimal commitment. 

---

Figure 1 The Iterative Development Strategy, Prototyping and Formative Evaluation
whilst enabling valid evaluations to be performed. An iterative development process
requires the use of evolutionary prototyping as a tool for exploration as well as
development. There are three basic types of prototype: Laboratory, Field and Delivery
Prototypes. Laboratory prototypes include screen representations and dialogue
simulations and may have a set of minimal functions but are not sufficiently robust to be
delivered to user sites unsupported. They are used by developers to evaluate technical
issues and also to carry out usability tests in experimental task scenarios. Field prototypes
can be delivered to the user work environment where they are used to evaluate the support
the system provides for real tasks and the impact this has on the general work design.
They need to be more robust than the Laboratory prototypes but may not necessarily
provide full functionality. They are used to validate the requirements specification in a
more realistic situation, run in parallel with existing methods of performance. Delivery
prototypes can be delivered to users to support real tasks, but that may not provide full
functionality. To be effective, all the prototypes must be quick to build, realistic to users
and evolutionary. The technology support for prototype construction is discussed below.

5.2.2.3 Evaluation

Evaluation may be classified as formative, measurement or diagnostic. Formative
evaluation is particularly important early on and may be used with user interface mock-
ups without real functionality. However, as calls to the functional units are developed
these can be included in the prototypes and incorporated in the evaluations. Later in the
project, evaluations will be more diagnostic or measurement-oriented as attention turns to
minimising usability problems and achieving desired levels of performance. At the end of
the day, the goal is to support the users' task performance and to provide a system that
they experience as usable. It is important, therefore, to focus on the users' experience of
their task and of usability [8]. The results of the evaluations may take the form of new or
changed requirements which can be immediately fed back into the design process. The
presentation of prototypes and design proposals represents a vital mechanism for
achieving a full and accurate understanding of the objectives of the system [9]. Methods
that support the different types of evaluation are essential for the evolution of the different prototypes.

5.2.2.4 Methods

Many methods have been developed to support different development strategies. There is a need for methods that can be applied at different levels of complexity, according to the needs of the problem domain and development context. No single method will be appropriate for all development contexts and there is a need for methods that already embody considerable tailoring so that they can be picked up and used by developers, once selected from a "toolbox".

The FOCUS methods address the need for 'discount' analysis methods. For that reason, they are quick to learn and easy to use for system developers. They may form the basis for evaluation reports or simply result in changes to a prototype leading to the next version. One example is the User-Software Observation Method (USOM) which involves users performing tasks and developers observing and discussing task performance with the users. It is an evaluation method which uses observational and verbal protocol techniques.

5.2.3 Front End System Architecture and Tools

A Front End System is a separable user interface system that integrates new and existing applications, services and knowledge-based task support in order to provide specific users with tailored solutions. FES Technology is based on an extended Seeheim model, with a distributed modular architecture that employs a client-server level of separation between the modules. It has a specialised Application Interface Module and a Support Module that addresses task support and highly interactive applications. The fundamental concepts were described by Edmonds & McDaid [10] and refined in Edmonds et al [3]. The major components and relationships within the architecture are shown in Figure 2 below.
The kind of modular architecture shown in the figure above can only work if communication between its components is simple, fast and flexible. FES communication is handled by 'messages', which are routed by the Communications Manager. The following sections briefly describe the concepts involved and the function of each component shown in the architecture.

5.2.3.1 **User Interaction Manager (UIM)**

The User Interaction Manager provides user interface presentation and dialogue services to the other modules in the architecture. In addition, it controls access to the user and mediates between modules in conflict for access to the user. Its main function is to create and realise the details of the object oriented dialogue specification it receives via the central message system.

The UIM is composed of three major components. The Dynamic Dialogue Manager (DDM) enables the monitoring of all dialogue traffic and where necessary the limiting of access to the user because, for example the screen is already very full and there is a clear risk of overloading the user. The Dynamic Presentation Manager (DPM) is responsible for the maintenance of the logical state of the interface and the interaction objects. It creates, updates and destroys Abstraction Interaction Objects (AIOs), using a library of AIO prototypes. When an event occurs at the end user interface the DPM sends an appropriate return message, using the standard message format. The Physical Presentation Layer (PPL)
handles those parts of the presentation function that are specific to a particular windowing system and tool kit. It is responsible for mapping instantiated interaction objects received from the DPM to the presentation system in use.

5.2.3.2 Application Interaction Manager (AIM)

If existing software is to be integrated to support a user's task, it cannot use standard messages to communicate. Therefore, these applications must be integrated into the Front End System technology through the Application Interaction Manager (AIM) [7]. The AIM consists of two main components. The Task Manager deals with abstract application independent tasks. It passes a request to the Application Action Manager which has within it all the information necessary to realise that task in an application specific manner.

5.2.3.3 Support Modules

A Support Module communicates directly with the rest of the architecture using messages. Support Modules can have a variety of roles, although they may be re-usable in different systems. They range from having general purpose dialogue models to having a task specific dialogue that involves an integration of dialogue model and executable task model. This integration can either involve a logical separation [12], or a physical separation [14] of these models. In the latter case, individual models 'communicate' using the standard message system.

Support Modules provide a range of functionality, such as analysis or complex knowledge base manipulation or visualisation tools, which are not available as Managed Applications. They typically support complex tasks and contains considerable knowledge based task and domain support. They involve task level integration of application functionality to support the user's task(s). They may contain an executable model of the task. There is an integration of the task and dialogue models so that each sub-task, or 'goal', relates to dialogue with the user to obtain data or decision information, internal domain specific processing or a task specified in the AIM's Task Manager. To obtain the data to satisfy its goals, a Support Module may, for example, obtain values for all the necessary parameters by
interacting with the end user (via messages and AIOs) and then send a message to the AIM containing an application-independent task description. Alternatively, it may access the AIM to obtain data to support communication with the user.

5.2.4 Technology Support to Strategy

The strategy and technology have been described in brief. It is, however, the combination that is significant. It is important that this combination is rather an integration and therefore, the integration needs to be of a reasonable depth to produce an effective solution. The section below discusses the integration between the technology and the strategy and methods.

5.2.4.1 Support for Iterative Development

The Front End technology directly supports the strategy in two main areas: those of Prototyping and Evaluation. In addition it provides considerable support for the project management which is important in making the strategy commercially feasible.

5.2.4.2 Support for Prototype Construction

The FES System technology provides a range of support to the incremental prototyping approach from laboratory to delivery prototypes The FES toolkit enables the developer to specify user-system dialogues much more rapidly than using conventional programming languages. This is because one can define a dialogue at an abstract level, leaving details of presentation to the User Interaction Manager to determine. Prototypes that simulate calls to underlying functionality that may not yet have been connected can be developed. This enables the FES technology to support rapid user interface prototyping. Also, when building a standard front end (i.e. to existing functionality), one does not have to implement the underlying functionality, but just to call it. In particular, it is much quicker and easier to alter such function calls than to change the actual functional code. These prototypes offering extensive functionality can still be changed with relatively little effort. For certain FESs there will be components that represent new functionality (complex Support Modules). These will have to be developed along more conventional lines but can be
integrated with the rest of the system when they are stable. This provides a prototyping tool kit of power and flexibility.

The FES AIO provides a rapid specification method with a limited learning curve and the use of defaults to further accelerate the specification process. Using the development tools for prototyping means that software redesign is only needed if fundamental changes are called for. The fidelity, scope and availability of the prototypes developed will also affect the validity of evaluation. Because the tools used are the delivery tools, the prototype will be faithful in 'look and feel' to the proposed system. The functionality and function calls that are implemented will call the Managed Applications. The prototype will increase in breadth as these function calls are developed. However, the functional components will only be integrated when they are stable, and, therefore, the user interface prototypes will be almost continuously executable.

5.2.4.3 Support for Evaluation

The FES Technology provides a range of support for evaluation. The two main areas are the provision of facilities such as user logging and the ability to generate changes to prototypes rapidly in a structured and well managed manner. Logging facilities provide records of user interactions. This facility is accessed via a simple high level switch. There are facilities to pre-process logs to produce manageable quantities of data. The ability to change prototypes quickly at a range of granularity in response to evaluation is of considerable importance. This can range from changing the detail of a menu item which involves simple text editing to rapidly creating new interface objects using the AIO specification method. The FES technology generates the interface 'on the fly' and therefore there are no detailed representations such as State Transition Networks to edit when changing dialogue details. Finally, the modularity of the system means that a whole new module can be developed in response to evaluation and easily integrated as the module interfaces are defined by the messaging system.
5.2.4.4 Support for Project Management

The modular construction of the architecture has important implications for the control and management of the software development process. Individual modules can be developed in parallel and various teams can be created to construct the different modules. The development task is, therefore, broken down into manageable development components supported by the underlying architecture. The minimal interfaces between the architectural components eases the management of the development process by limiting the interdependencies that have to be managed.

The technology, strategy and methods have been used to develop various Front End systems one of which SEPSOL [14], a complex multi-application system that supports chemists using statistical design to identify suitable models for their experiments. It employs considerable domain knowledge in a variety of roles. The support provided by the technology and methods enabled the completion of SEPSOL within the target time of twelve person months.

5.2.5 Conclusion

This paper has described an approach to Front End System development that involves the combination of a development strategy and a supporting technology. It has described the main points of the strategy and the key components of the technology.

References


3. Edmonds, E.A., Murray, B.S., Ghazikhanian, J. and Heggie, S.P. The re-use and integration of existing software: a central role for the intelligent user interface. People


* * * *

The work described in 5.2 was expanded and published as follows:-User Centred Complex System Design: Combining Strategy, Methods and Front End Technology (Murray Candy
The following describes the methods that were developed and made available to companies. Methods for user centred system design have been devised to support different development strategies. From collaborative work in industrial and commercial environments, we have identified a need for methods that can be applied at different levels of complexity, according to the needs of the problem domain and development context. No single method is likely to be appropriate for all environments and therefore, the methods have been tailored and tested in real use and are available to developers as "toolbox" selection. Methods must also offer effective support for user centred analysis but at a lesser cost than existing methods. In that respect, they should not be perceived as demanding extensive investment of time and effort. They should, however, increase the focus on the users' characteristics, task and requirements, by encouraging the identification of key areas of information that need to be collected and providing useful notations for representing the information.

A number of existing methods were researched and assessed in relation to the requirement for cost effective and practical resources for use by system developers. The following methods in particular were drawn upon: HUFIT PAS toolset (Taylor, 1990), the USTM methodology (Macaulay et al., 1990), Contextual Inquiry (Wixon et al., 1990) Hierarchical Task Analysis (HTA, Annett and Duncan, 1967) and Task Allocation Charts (TAC, Ip et al., 1990). They were customised and tested in the design and development process of a number of FES systems. Modifications and improvements were then carried out in the light of practical experience.

The FOCUS Requirements Analysis and Prototype Evaluation methods were designed to meet a set of basic requirements. In particular, they address the need for 'discount' analysis methods (Nielsen, 1989). For that reason, they are quick to learn and easy to use for system developers. They are also designed to deliver results from relatively short contact with representatives of the user population. They may form the basis for evaluation reports
or simply result in changes to a prototype leading to the next version. A suite of methods was developed for FES developers to accompany the FOCUS toolkit (Candy and Rousseau, 1995).
6.1  INTRODUCING AN EXPERT SYSTEM INTO AN OFFICE ENVIRONMENT: USERS, EVALUATION AND THE DESIGN PROCESS


The paper describes some practical experiences in the development and introduction of an expert system into an office environment. A number of points are raised which offer one perspective on the way Human Computer Interaction issues can be brought to bear on the design of expert systems. In particular, the role that evaluation from a user perspective can play in the design process is considered.

Keywords: Expert system, user interface, user requirements, evaluation, design process.

6.1.1  Introduction

An application is under development in which expert system support is being introduced into an office environment in an engineering company. A broad aim of the application is to demonstrate the use of logic programming integrated with existing conventional database systems in a business environment (Nomura and Lunn, 1987). The organisation of the work is on a collaborative basis: the partners consist of a software company, Telecomputing plc, the lead partner and the provider of the application software, the Loughborough University of Technology Computer-Human Interface Research Centre whose role is to build the user interfaces for the expert system and to consider the broader aspects of system development, Brunel University and a manufacturing company for which the application is being developed.

The expert system is intended to provide support for estimators in an office environment in the engineering company. The Estimating department devises estimates of manufacturing costs for new parts prior to tender called 'bids' or 'projects', set down in the form of an operation layout or set of total machining actions. The advice provided by the system is based upon the knowledge of experts in the field and will be used, in the first instance, by experienced estimators.
The application of expert system technology to the task domain in question was proposed because the knowledge required was considered to be of a sufficient degree of difficulty and not amenable to conventional programming methods. In addition, an important aspect was that it afforded the possibility of a link to conventional data and, hence, of demonstrating integration.

From the engineering company's point of view, there was an imminent shortage of experienced personnel in the field and concern to provide a means of training to bring staff up to a satisfactory level of expertise. Estimators are normally recruited from the shop floor and receive training by working alongside experienced personnel. All must have a basic knowledge of company plant and machining processes on which the estimating activity is dependent. The process of becoming an expert estimator takes from between three and five years.

The first design concept arising out of the early knowledge acquisition work was of a co-operative problem solving system whereby the estimator specifies relevant part and material information which is used to generate menus of machining actions (Slatter, Nomura and Lunn, 1988). When the user has completed the entry and selection process the system can provide a final check of the whole estimate.

The target of the system support, the final writing up activity, takes place towards the end of the production of a bid estimate from which point the senior management have available the material on which to present outgoing estimates. However, this activity in itself was not considered by the estimators to be the critical point in the whole process: by this time, the necessary information has been gathered and all the important decisions have already been taken. Thus, the system design was not based upon the operational methods of the people who were to use the system but was, in effect, more useful to management. The knowledge acquisition concentrated on identifying information about part type classifications, materials categories and machining operations and their inter-relationships. The importance of other activities, such as information gathering, consultation with other departments and constant referral to the scheme drawings, to the estimators working
practices was consequently under-rated or not understood.

A different perspective on the system design was provided by the user interface team whose focus of attention was on user task scope and how the user interface design might support that process. Interviews conducted with the estimators as preparation for the evaluation exercise, revealed that there was no requirement for support for the writing up aspect of their work but a need for on-line information. It also emerged that the nature of the support required reflected in the first prototype, had indeed come from management whilst interviews with the estimators had been mainly concerned with the foundations of estimating and engineering knowledge, domain knowledge which excluded the methods used in the application of that underlying expertise. The possibility of conflict in end user and management requirements was not addressed in the early stages of identifying system functions.

6.1.2 The Initial Design Process

6.1.2.1 Knowledge Acquisition

The history of the development was as follows: a team to carry out knowledge acquisition was formed within the software company. That knowledge acquisition team led the early activities and preceded the user interface team into the user site. The intention was to install a prototype which would lead to incremental change towards the final product.

The procedure was a phased, incremental one, as follows:

.........feasibility study
    knowledge acquisition
    functional specification
    user interface
    implementation
    evaluation
    ....modifications
    next phase of design
    product
This process broadly conforms to accepted standards and methods (CCTA, 1985). The feasibility study was the first step. This was followed by two main investigatory activities:

1. Business analysis, organisational requirements - investigation of information resources - existing databases and manual systems. (Stow, Lunn and Slatter, 1986).
2. Knowledge acquisition-classifications of part types, categories of materials, machining operations, and their interrelationships.

The user interface team had doubts about the appropriateness of this process but the outcome in relation to user acceptability was not clear in the early stages. There was no early explicit documentation of end user operational requirements, reflecting a particular knowledge based focus, rather than one based upon end user requirements. A formal theoretical view of the estimating process was taken, rather than one based upon empirical evidence.

6.1.2.2 The User Interface

The issues surrounding the approach adopted to knowledge acquisition can be illuminated somewhat by a consideration of such systems from a user interface perspective.

The user interface can be thought of as having three levels (Edmonds, 1982; Green, 1985). The first, and outermost, level is the presentation manager which is concerned with how information appears on the screen, in terms of organisation, type fonts, colour etc. and with the specific actions taken by the user: i.e. which button is to be depressed and so on. It is, incidentally, not uncommon to see this level as the totality of the user interface and, hence, to minimise its importance at the early stages of design.

At the next level is the dialogue manager, which determines how the user moves between menus, for example. It controls the dialogue between the user and the rest of the system, allowing levels of freedom and constraint for the user: the modes of the user interface.

At the third level is the application model, which provides an abstract description of the objects and actions of the application, expressed, for example, in Prolog structures.
The description of the functions of the system resides at the application level and so is closely bound up with the functional user requirements. The operational user requirements, however, also include many issues that only find expression at the dialogue level and even the presentation level. The application model level cannot be determined entirely in advance of the other levels of the user interface because operational issues are as vital for success as the obvious functional ones. This view conflicts, however, with the initial design and development process employed in this project and we will see that the process was changed in the light of experience.

6.1.3 Evaluation of the Prototypes

6.1.3.1 Aims

1. To evaluate the prototype system in terms of its suitability for the task on hand from the user perspective.

2. To provide feedback on the design and operation of the system or the next stage of design and implementation work.

The investigations covered the following areas:-

- Design features of the user interface
- Accuracy and appropriateness of the knowledge base
- Reliability, performance and usability of the whole system
- Acceptability and ease of use of whole system to users

6.1.3.2 Data Collection Methods

1. Interviews with prospective users took place prior to installation of the first prototype to identify existing working methods and those issues significant to both the tasks in hand and the expert system use. Structured interview formats were devised on the basis of taped pilot unstructured interviews.
2. In the prototype installation period, formal, structured interviews were conducted with end users. During initial training, informal discussion took place about the system use and related activities. The users were encouraged to record their observations about the system use as a separate activity to making notes on the system 'scratch pad'. In addition, the automatic time stamped recording of the use of the user interface took place.

6.1.3.3 Criteria for Evaluation

The intention was to cover the areas listed below in as much detail as possible. In the event, severe operational difficulties prevented a detailed investigation of all of the topics and full use of the data collection methods.

1. General Issues
   • user functions: full task scope
   • user perceptions of system use

2. The Knowledge Base
   • accuracy, organisation and presentation of knowledge;
   • appropriateness for the task

3. The User Interface
   • dialogue design; style of interaction
   • functions and interaction routes
   • screen design and layout

6.1.4 Evaluation of the User Interface

6.1.4.1 Separation of the User Interface

Technical difficulties delayed the installation of the first prototype system. This led to the notion of installing the user interface alone as the first stage of the user evaluation. The evaluation of the stand alone user interface which followed, primarily involved end users. Their reaction, faced with the first direct experience of the projected system, confirmed initial doubts about the matching of the design to the end users' needs. The possibility that
the users might not use the system was then apparent to the whole team. A reassessment
was made when it became apparent that the users were likely to reject the system, leading
to new methods, as described below.

The decision to install a stand alone user interface made it possible to evaluate that
component separately from the knowledge base. One advantage was the early identification
of issues requiring change in the user interface. It was, thus, possible to carry out a limited
number of changes prior to the installation of the whole prototype system. With the
linking of the two components, the identification of those design functions and features
attributable to the expert system and the architecture as distinct from the user interface was
possible.

It also provided an opportunity for the users to become familiar with the software and for
the development team to gain immediate feedback about its use and appropriateness. That
there would be differences when the full system was installed was explained. This would
take the form of menus of suggestions on the operations to be carried out on the basis of
the information supplied by the user and checks from the system on accuracy and
omissions. The end users were then interviewed and asked to complete a questionnaire.
Informal discussions took place with management.

6.1.4.2 Results

As a result of this evaluation exercise a number of changes were made to the user interface.
The changes made were of limited impact on the design of the user interface. They ranged
from alterations to wording and type of input required to modifications in the specification
of machines from named machine to machine groups. A number of items were considered
to be somewhat unsatisfactory but were not changed for the prototype installation.

In addition to these specific user interface items, the evaluation led to a reconsideration of
aspects of the whole system and of the design and development process being employed.
An issue which emerged more explicitly included the difference of perspective between the
end users and management. The latter had a more positive view of the envisaged system,
whilst, nevertheless, wishing to see details altered and more training to overcome initial problems. Whilst management was mainly concerned with the details of screen design and selection methods, the prospective end users did not see how they could usefully incorporate it into the estimating task because it focussed on the final and, from their perspective, least significant part of the process. The general view was that the method used was likely to slow them down because of the amount of typing involved and the lack of flexibility. It was clear that the performance of the system was critical in a number of respects, not least of which was the accuracy and completeness of the advice from the expert system, which can reduce the need for user actions. Up to this point, no measures had been placed against those items in relation to user acceptability.

6.1.4.3 A Design Process Re-Structured

The evaluation of the user interface was instrumental in making explicit issues concerning the design of the whole system, as described above, and it became clear that the process used had allowed some significant matters to remain implicit for too long.

The immediate outcome was a change in project organisation and procedures. The design process adopted is comparable to a project reported by Eason and others, 1987. It differs in placing more emphasis on end users and the user interface early on. The application development teams are now inter-establishment, divided into design and technical rather than user interface and knowledge base. A member of the user interface team co-ordinates the design team and a representative from the user site co-ordinates the technical team. The first cycle is in process and a new design for the base system is nearly complete.

6.1.5 Evaluation of the Prototype System

A prototype system with all components was installed and evaluated in a similar, but extended manner to the stand alone user interface. Some of the results are briefly described below.

In the context of the whole end user task, this expert system is placed at the end of the process. There is no support for the drawing analysis and information gathering components of the process. A number of aspects of the system were called into question,
in particular, the way that the user must follow a tightly constrained formula in the development of some of their work.

6.1.5.1 Reliability and Performance

The problems experienced in the evaluation of the first prototype highlight some issues regarding the approach adopted. A lack of robustness and reliability in the system influenced the results of the evaluation in two ways:

1. it prevented the users from carrying out an extensive and thorough evaluation of the knowledge base.

2. it affected the users' attitudes to the system as a whole and, by implication, their confidence in the development team.

The experience argues for the use of more extensive limited function prototyping as was carried out in the user interface evaluation and a technical comparison of the expert system output with existing evidence.

Another aspect, mentioned above, was that the performance of the expert system, in the sense of its level of expertise, the accuracy and completeness of its advice, was critically important and needed to be addressed within user requirements.

6.1.5.2 User Acceptance

It had been established in interviews conducted prior to the installation, that most prospective end users had either no previous experience in the use of computers or a limited one. Whilst there was no particular antipathy to computer use, there was no positive call for it either. There was some scepticism that bid estimation could be addressed adequately in a computer system. The interviews also indicated that, in relation to estimators' task scope and information gathering activities, the prototype system did not address those areas that were considered by them to be of primary importance. On the other hand, management requirements were different and, as a consequence, so was the
perception of what constituted suitable system support.

6.1.5.2 Impact on Estimating Task Scope

The prototype is directed towards supporting an activity which is the final stage of a skilled mental and visualisation process. It does not support the important prior drawing analysis stage or the gathering of information necessary to the specification of the operation layout. Tests suggested that without the incorporation of drawing information, either within the expert system or as a specifically anticipated integrated manual activity, it is impossible to arrive at a detailed, accurate estimate.

6.1.5.3 Relationship to Current Practice

The immediate factors to consider are those which directly affect the specialised procedures of the expert at work. The modus operandi of the prototype system conflicts with existing practice in a number of ways and to different degrees, according to the individual estimator:

Implicit in the current design is a pressure on the end users for standardisation of procedures and terminology. It has implications for training and is a matter which must be addressed in the early design stages of the system.

There are a number of mutually exclusive stages of manufacture built into the design. This is one factor in the generation of menu items. All such groups of items refer to a particular stage. To insert operations within stages accurately the estimator needs to have knowledge of the concept and how it operates within the system. There is limited opportunity to move between stages. It is possible to move forward a stage but not backwards. The estimators were unfamiliar with the concept of these stages of manufacture and were not forthcoming about its value.

Any computer support to existing working practices will introduce change of one form or another. The true extent of that change and its likely impact can only be judged at the time of its introduction into practice. However, it is clear that the effect on people's attitudes
and the successful integration into practice will be influenced by the manner of its introduction, including prior involvement in design and development (Damodaran and Eason, 1981; Mumford, 1983). It is also reasonable to expect that factors such as training, time allocation and environmental resources may play a part. Whilst it is natural that the introduction of a new system may alter practice, it is important to be clear about such changes in order that deliberate decisions are taken and appropriate training is provided.

6.1.6 Discussion

The design and development of an expert system, particularly for a bespoke application, is difficult. There are reasons for this which arise from the general context of computer system design. Traditional approaches to expert system development place end user requirements below and behind functionality and domain knowledge acquisition. The expectation often seems to be that technology should provide the impetus and its advance is, in itself, the main goal. There are also unrealistic expectations from users and developers alike, some of which have been fuelled by wild claims about the potential of expert systems.

A number of conclusions can be drawn from the experience of this project. Klein and Newman (1987) refer to the 'different cultures' of user interface and expert systems developers and to the difficulty of obtaining recognition for the importance of the user interface. In applications which so clearly impinge upon the working practices of experts, it is a cultural gap which must be bridged if the systems are to be acceptable and useable. It is essential for designers and developers to understand the need to:

- place user requirements first from the outset;
- recognise that there may be different classes of users of the same system with different requirements;
- understand that there will be changes in user perceptions during the design and development process, and, most important, that expert user requirements are likely to be complex (Edmonds, 1987).

There is a need to provide a means to facilitate requirements formulation and, to this end,
prototyping is essential. As Lansdowne (1987) says, "it is virtually impossible to articulate needs independently of the means of meeting them". It is important that early prototyping should not be seen as the first steps in incremental development because that would suggest that a fuller understanding of the needs could be obtained independently of the means than is likely to be the case. Thus, evaluation in the field has a relationship to the clarification of user requirements. It was clear that user requirements analysis, prototyping and evaluation are interrelated and that this must be recognised if there is to be more successful system design and implementation. In particular, in the case of expert systems, it is neither feasible nor practical to attempt to determine the full scope or degree of completeness of the knowledge required before initial prototyping and evaluation have taken place.

6.1.7 Conclusions

It is clear that user related issues are particularly significant when introducing an expert system. The study suggests that the design process should include in its early stages a careful analysis of the total scope user needs and requirements, assisted by prototyping the user interface as a primary mechanism. Certain questions would appear to be particularly important in an application development such as this. They include the following: what is it about the task that is difficult? who in the organisation is able to judge that? what is the scope of the expertise that is required? what constitutes beneficial assistance? The last point is particularly important. It is clear that an expert system, like any other piece of software, must perform to an appropriate level in order for it to satisfy user requirements. There is a need to explore this question and to start to set performance measures in relation to the embedded knowledge early in the design and development cycle as part of the requirements specification. There is an urgent need to explore appropriate ways of doing this effectively. Precisely how, remains a research problem.

References


7.1 END USER MANIPULATION OF A KNOWLEDGE BASED SYSTEM:
A STUDY OF AN EXPERT'S PRACTICE


The advent of End User Manipulation of Knowledge based Systems (EUKMS) provides new opportunities for addressing the problems of encapsulating domain expertise. Interfaces which enable the expert, a professional and/or scientific practitioner, to create, refine and evaluate rules about the constituent elements of their knowledge, provide a means of circumventing some of the current barriers to successful knowledge encapsulation. The critical feature of the design of such systems is the provision of facilities for the automatic conversion of the expert's rules into code. In a study of scientific work involving the capture of phonetics expertise in a knowledge based system, key aspects of a speech scientist's working practice were identified. This paper discusses that use of the Speech Knowledge Interface system (SKI) in the context of investigations into the construction of an enhanced model of speech production for a speaker independent, continuous speech recogniser. Evidence that providing the expert with an appropriate interface to a knowledge based system stimulates questions about existing knowledge and gives rise to new insights into the scope of the investigations, was found. Thus, the process of knowledge externalisation, both of knowledge which was only partially realised and knowledge that was perceived as "new" by the expert, was facilitated by the interaction with the system.

Keywords: Speech Science, Spectrogram Reading, Action Research, Knowledge Based Systems, End User Knowledge Manipulation, Knowledge Externalisation, Knowledge Encapsulation

7.1.1. Introduction

A study was conducted into a scientist's investigations in knowledge-based speech recognition using an End User Knowledge Manipulation System (EUKMS) (Edmonds, McDaid and Bayley, 1991). This paper reports the approach adopted and techniques used, including a brief description of the the system employed, and the results of study of the user-system interaction.
interaction. The scientific context was the construction of an enhanced model of speech production intended to contribute to the design of Continuous Speech Recognition (CSR) systems and to the body of knowledge necessary for speech synthesis-by-rule.

The interpretation of the speech signal is complex and even a matter of dispute amongst experts (Bladon, 1986; Bailey and Summerfield, 1980). Nevertheless by employing end user knowledge manipulation techniques it was demonstrated that it is possible to encapsulate relevant knowledge in machine usable form (O'Brien, 1989 a & b). The Speech Knowledge Interface (SKI) (McDaid and Edmonds, 1990) was developed as a method for enabling the expert to interact directly with domain specific objects and to express and encapsulate knowledge about them. The enabling conditions for this method of knowledge externalisation and encapsulation are described. The immediate goal of the study was to acquire an understanding of expert practice that could be applied to the design of a computer system. To that end, it was conducted using a participative approach that included the expert's observations as part of the evidence gathered. Key aspects of the expert's practice were identified and grouped into three main categories, Operational Issues, Knowledge Externalisation and Knowledge Encapsulation.

The role of the system in prompting awareness of implicit or 'tacit' knowledge and its application to the generation of rules proved to be important. It was observed that the process of knowledge externalisation was facilitated by the interaction with the system. Some of that domain knowledge had previously been only partially realised by the expert. Thus, new understandings about the structure and content of this knowledge being applied emerged. Whilst the initial design of the system satisfied the expert's requirements, as experience led to more demanding investigations within the domain, its limitations for that work became apparent. In effect, the expert's task scope had been transformed and the net result was a requirement for more control over the structure of the knowledge base. A broad framework for establishing design goals arising from this approach to The role of the system in prompting awareness of implicit or 'tacit' knowledge and its the capture of domain expertise is discussed.
7.1.2 Knowledge Encapsulation by Domain Experts

The design and implementation of graphical programming systems (Murray & McDaid, 1993) that have encouraged the manipulation of knowledge bases by end users is a growing area of research (Tuhrim et al., 1988). The problem of the acquisition and encapsulation of expert knowledge is addressed by the use of graphical interface techniques. The direct interaction with a knowledge base using domain specific representations of the source data, and without the need to program, is a process that transforms the nature of the knowledge externalisation and its capture in machine useable form. We are becoming familiar with the notion of an expert externalising and refining existing knowledge in this way but are less aware of the potential for the creation of new knowledge that this method affords. There are fundamental questions raised when a domain expert interacts with a knowledge-base. We have shown that it is not sufficient for a system merely to support the expert in encapsulating domain knowledge: the use of a knowledge based system changed the expert's working practice from which new requirements and new concepts of the domain knowledge emerged.

Knowledge-based systems for Continuous Speech Recognition (CSR) have had limited success because of the inherent difficulty of acquiring and encapsulating the domain expertise; see, for example, Memmi et al. (1984); Johnson et al. (1985); Stern et al. (1986); Carbonell et al. (1986); Conolly et al. (1986); Lamel (1988). Typically, CSR systems model spectrogram reading in the search for the appropriate cognitive models that underpin expert performance. This presupposes that expert knowledge to support the modelling task is already available. This cannot be assumed for two main reasons: firstly, an expert's knowledge may not be readily accessible or in an appropriate form and, secondly, by the very nature of the expert's activity, it is partially formed or incomplete because it is in the process of being developed. In these circumstances, it is important, therefore, not to conflate systems for cognitive modelling with the parallel need to provide, in another kind of system, a mechanism for the expression of domain knowledge.

A key problem arises from the fact that experts are often not able to state explicitly what the relevant knowledge is even though they might be able to exhibit in practice an ability to use it (Berry, 1987). The SKI system was developed as a method for enabling the expert to interact
directly with domain specific objects, (e.g. the spectrogram) and to express and encapsulate knowledge about them without having to use a programming language directly.

7.1.3 The Speech Domain Exemplar

The expert user's investigations of rules to identify a subset of the speech sounds of English in the construction of an enhanced model of speech production was the subject of the study reported. The investigations attempted to formulate and evaluate a set of relevant cues or features to be used to distinguish the sounds according to identity. A number of rule bases were developed from an existing core rule base and a study made of the contribution of the cues to the identification process using a set of source data (O'Brien, 1989 a & b).

This work built upon earlier knowledge-base speech research in which a method was sought whereby a best match between expert human knowledge and a machine-useable representation could be achieved. For this purpose, the spectrogram, a visual representation of the speech signal, was selected (Johnson et al, 1985). The formulation of rules about speech knowledge in the SKI knowledge based system was based upon an analysis of the spectrogram source data. The point about this representation is that it is directly meaningful to speech experts as well as being available for automatic analysis and visual display.

The strategy adopted was to work from humanly perceived patterns towards the identification of objective definitions for computer readability. The specific advantage of the use of the spectrogram is that experts are able to identify direct relationships between its features and the sounds uttered without knowing the identity of the speaker (Zue & Cole, 1979). It was important to retain this advantage and, at the same time, use it as a vehicle for computer recognition. In this way, a direct relationship between the source data (the spectrogram) about which the human expert applied and generated the knowledge and the machine-readable formalism was achieved.

7.1.3.2 The Speech Knowledge System

The Speech Knowledge Based Interface (SKI) was designed and developed based upon results from earlier work into the construction of a speech knowledge base (Connolly et al, 1986). The
significant difference was that the speech expert was provided with graphical methods for interaction instead of having to program. The main facilities were:

1. graphical techniques for 'annotating' the spectrogram features
2. direct access to the construction, testing and refining of rules specifying which speech events have occurred given the features identified in i) above.

Both the features and rules are translated into a Prolog representation which is hidden from the user. A fuller description is to be found in McDaid et al (1991).

The user begins by annotating the spectrogram: in the first instance, this involves dividing it into segments that represent significant changes in the levels of intensity (reflected in the degree of darkness and light). The segments are then described in terms of a basic set of visual features, such as "silence", a white segment, or "fuzzy" a very dark one (Figure 1). From this point, the application of knowledge is expressed in terms of rules applied to the segmentation features. Rules are created by the user typing text into the three segment window which is divided into **previous**, **current** and **next** segments, each of which is further divided into "found" and "not-found", enabling knowledge to be expressed about the visual features in the speech signal. The vertical bars between segments in the rule-window translate as Prolog statements, horizontal lines between the text signify **and**, and bars signify **or**: (Figure 2): thus in the example:-

```
if the previous segment is not context initial, and
if the current segment has the to_p silence and is less than 40ms long and
if the next segment contains the event frication and is less than 40ms, and this event is not context final
OR
if the current segment has the to_p silence and is between 50-100ms long, and
if the next segment has the event frication and is between 40-100ms long
OR
if the previous segment is an /s/, and
if the current segment has the to_p silence and is between 50-100ms long, and
if the next segment has the event frication and is less than 40ms long
```

etc then return an event voiceless plosive.
Figure 1: SKI Screen with Annotation of Spectrogram

Figure 2: SKI Three Segment Rule window
Thus, the SKI methodology represents a convenient way of exploring knowledge-based speech investigation for the non-programmer. The rules are represented internally in Prolog without the need to acquire the considerable expertise otherwise required in that language (Edmonds, 1989). The speech scientist's goal was to arrive at definitions of suitable descriptors which could be objectively defined and at the same time were meaningful to an expert, and to use these as the input to a modified rule-base. Additional facilities which could be accessed simultaneously with SKI were the Speech Tools which provide supportive instrumental analysis consisting of different perspectives of the same data in the form of the low frequency energy, the wave form, the zero crossing rate displays.

7.1.4 Action Research for HCI

In the field of Human-Computer Interaction, the generation of new approaches to empirical methods is illustrated by the current interest in broader-based studies, for example, contextual research (Wixon et al., 1990). Challenges to mainstream methods have arisen because traditional experimental ones have proven to be inappropriate for the generation and testing of hypotheses involving complex human behaviour. Some have argued that the particular case of rapidly emerging technologies demands a radical reappraisal of the relationship between theory and practice (Carroll & Campbell, 1989; Carroll et al. 1990). The notion of the embodiment of theory in the designed object (i.e. the system) is analogous with earlier thinking about the actions of practitioners as 'theories-in-action' but ones that are, in effect, hidden assumptions that need to be uncovered and made explicit (Elliott, 1985; Schon, 1983). These approaches also relate to research that includes subjective data from the objects of study and attempts to apply the results directly to the ensuing actions, e.g. participative design (Mumford & Weir, 1979; Mumford, 1981).

In order to gain a better understanding of users, technology and tasks and the inter-relationship between the different elements, it is necessary, in the first instance, to adopt an open and exploratory approach. Where the goal is to arrive at better system design through application of the results of such study, it is essential that the knowledge obtained is both practical and applicable. To that end the approach adopted in the study was based broadly on the 'Action Research' model. It is one that is not novel, having emerged from the pioneering work of
Dewey and Lewin (Dewey, 1929; Lewin, 1948, 1951) and followed through in a number of different domains, mostly particularly education (Candy, 1988).

Action Research has been adopted where the goal is to apply research results directly and immediately to practice. Three distinct elements of action research were incorporated in the approach used as follows:-

1. that the knowledge to be acquired should include the perceptions of the prime practitioner,
2. that there should be active participation by the investigator rather than adopting an external observer stance,
3. that by introducing change into a situation one can better understand it, i.e. by introducing a computer system into working practices we can identify the critical issues both for the working practices and the design of the computer system.

The collection of empirical data in the field and giving close attention to specific events within a whole context rather than the isolation of variables are characteristics of action research. The investigator begins by gathering details about the broad context of the domain expert's activities and takes a pro-active role in the process of eliciting information. Any data or analysis arrived at by the investigator is referred to the expert for comment. This, in effect, becomes a partnership, rather than an observer-subject relationship. The outcome of such a process is, in that sense, the product of a joint effort and does not solely assume the character of the investigator's observations. It is a form of knowledge acquisition but one where the aim is to acquire a broad range of contextual data without imposing a pre-conceived theoretical construct upon the knowledge.

A prior study (Woodcock, 1986) investigated the process of spectrogram interpretation by conducting a protocol analysis of verbalisation during reading sessions. The goal was to produce a representation of the way the expert carries out the task. The study was one which looked at the expert's task from an external observer stance, and as such paralleled similar activities in other groups (see eg Carbonell et al., 1986; Stern et al., 1986), a common aim being to understand the underlying thought processes and to arrive at a formal description of the expert's reasoning. If an adequate cognitive model could be deduced then this would contribute
to the development of a reasoning framework to support the expression of complex knowledge. However, the results proved inconclusive and raised questions as to whether it is possible:—

1. to understand such issues without including the perceptions of the subject/expert in the investigations
2. to elicit the implicit knowledge which the expert does not articulate because it is 'obvious'
3. to discover whether an implicit hierarchy of information underlies the expert's decision-making, and to investigate those events which prompt its restructuring, with cues losing or assuming primary or secondary status
4. to prompt the expert to an awareness of 2. and 3. and make them explicit in a machine-robust manner.

In the study reported in this paper, the goal was to provide a means whereby those aspects of the goals, methods and working practices of an expert are revealed. Cognitive processes were identified and expressed as goals and intentions, procedures, extensions to and influences on knowledge and its application. These are not normally revealed by an analysis of a task-based simultaneous verbalisation. The key elements that informed the work under discussion were the scientific task context and the expert practitioner's observations and reflections upon her work.

The first stage of the study began by close monitoring of the speech expert at work. The expert also monitored her own activities, regularly reviewed the previous tasks, assessed their significance and documented them in periodic status reports which were then used as a further basis for discussion. In the second stage, these issues were then examined more closely by direct observation and video recording of 'live' speech investigations i.e. actual work in progress, not activities devised for experimental purposes.

The investigations revealed that a difficult step proved to be bridging the gap between low level articulation at the task execution point and higher level abstraction from the detailed examples into more general principles. Here, the role of the investigator was to identify questions, connections and relationships based upon that data collected and then engage the
expert in a discourse on these issues, i.e. promote a higher level analysis. This process is made easier by the readiness of the practitioner to reflect upon the work in hand (Schön, 1983). The outcome of this study was the result of a synthesis of the expert practitioner's observations and the investigator's identification of issues which arose from those observations.

In the earlier work of Woodcock, referred to above, it was thought that making verbal commentaries did not conflict with the spectrogram reading activity. We found, however, that making comments on what was happening when using the system created a conflict for the speech expert: in effect, it altered the way she carried out her work and interfered with the deliberations and decisions. As a result, in the second stage of the study, the computer based activity and the commentary on this were carried out separately. Video recordings and taped commentaries were made separately: the speech scientist commented on the annotations and rule base development in progress and reflected, where relevant, on the implications and issues arising from this data analysis. It is interesting in relation to this problem, to note the experiments of Broadbent et al. (1986) on the relation between the ability of a person to carry out efficient action in a situation and that of the same person to answer questions about the situation. The results suggest that the processes are specific to one or other mode. There are alternative modes of processing in human decision making, each mode having particular advantages.

7.1.5 Key Aspects of an Expert's Practice

The first set of results from the exploratory study indicated that the role of the system in prompting awareness of implicit knowledge and its application to the generation of rules was likely to prove important. There was a recognition by the expert of the way use of SKI encouraged rigour. The development of a robust model of speech production determined the boundaries of the expert's work and, thereby, influenced the manner of working. The impact of the computer based speech investigations on the expert's view of the reliability of existing speech knowledge was a noticeable outcome. There was an increase in scepticism about the universal applicability of existing models of speech production and perception and a realisation that although theoretical models were well attested in laboratory experiments with balanced materials, they could not necessarily be validated in the case of continuous speech.
From those findings and confirmation from the results of detailed observations and interviews, key aspects of the expert's practice can be categorised as follows:

- Operational Issues
- Knowledge Externalisation
- Knowledge Encapsulation

A full account appears in Candy and O'Brien (1989).

7.1.5.1 Operational Issues

In relation to the interaction with the spectrogram source data using SKI the following aspects were significant as follows:

- being able to have "holistic" and alternative simultaneous views
- being able to change procedures and actions at any time
- a relationship between perception and action

Views of source data

The ability to view and manipulate the whole spectrogram at once through out the annotation process was an important advantage offered by the SKI system.

There were, broadly speaking, two stages in the annotation process using SKI:

1. the "holistic labelling" at the first pass segmentation when the overall types of pattern are viewed, divided and marked very quickly in order to reduce the opportunity to make natural class judgments, followed by,

2. the 'analytical labelling' stage i.e. focus on detailed phenomena and the application of different levels of information to guide judgement.

It was most important that a complete view of the data was available. However, it was difficult for the expert to hold in her head all the information about parts of the data not being directly addressed at any one moment. Such data appeared to be relevant however, although it was not possible to predict in advance exactly what. It was, thus, important to provide methods for seeing all of the data at any one time and for viewing it in whatever form or forms

234
were desired. In this case, the data is very complex and the simultaneous availability of alternative ways of representing it proved invaluable.

The relationship between being able to observe the source data and apply, text and amend the rules under development was important and felt to be a necessary facility. In fact, multiple views of the source data in combination with the rule creation tables were typically displayed up to seven windows at a time without problems.

**Procedural change**

Three categories of changes to procedure were identified:

1. variation in annotation order,
2. delayed or reserved annotations and
3. unplanned alterations.

Prior to the study reported, the order in which the spectrogram annotation was carried out was described by the expert as a fixed order of working as follows: segmentation, voice bars, formant patterns, stripes, intensities, formant lines, silences, fuzzy cut-offs and compact/diffuse descriptors. This order was thought by the expert to be driven by the system design but a reassessment was made when attempting to arrive at a description of priorities used in annotation, i.e. a hierarchy of cues. The reassessment was that the order adopted was not, as was previously assumed, ad hoc and system driven, but task driven and logical in terms of the speech signal analysis process itself. However, the option to alter that procedure was considered to be essential.

Changes in order arose for different reasons occur during annotation. For the segmentation of the spectrogram, the usual order of completion is to partition left to right in a quick, fluent motion. The speed and fluency of the segmentation varied according to the perceived difficulty of the signal and this was sometimes manifest in a reversal of the order half way through the segmentation. It is clear that the procedural order was dependent on the degree of difficulty encountered and reflected uncertainty about how to mark some visual features.
In a number of cases, the annotation of particular segments was not carried out in the usual manner and order. This was, in the main, the result of doubts about the nature of the visual evidence and insufficient information from the other aids (Speech Tools). Uncertainty about what action to take with regard to particular features of the data is, of course, inherent in the task and, therefore, occurred frequently throughout most of the recorded annotations. Those spectrograms considered to be straightforward were in a minority. The response to such uncertainty was mainly not to act but to reserve judgment indefinitely or to await further annotation and any clarification that that might bring.

**The relationship between action and perception**

A significant finding was that the way in which the source data was annotated had an effect on how it was perceived. This included the speed with which it was carried out, particularly in the initial segmentation stage, the type of event and the order of annotation. An important feature of the process was the tendency to reassess the annotation after an intervening period of time and, also, when at further distance from the screen. In addition, drawing lines using a mouse and tracing over visual features led to reassessment of the significance of those features in relation to the underlying speech events. The assessment and manipulation of the source data raised important issues in relation to the handling of visual information by the expert. Having set out, in the first instance, to exclude factors which might influence the segmentation decisions beyond those based on the visual evidence, it was clear that the direct marking of the data introduced new elements into the way it was perceived. The influence of action on perception is clearly a significant issue.

**7.1.5.2 Knowledge Externalisation : From Implicit to Explicit Knowledge**

The capacity of the expert to reflect on her practice and to extend her understanding of existing knowledge is an attribute which proved to be a significant aspect of this study. This enabled her to develop greater awareness of the underlying processes at work in the application of her knowledge to the task in hand. Two distinct but complementary factors were observed in the process of reflection and knowledge externalisation by the expert, each having different outcomes.
The participative research method itself was a factor in acquiring evidence about the broader aspects of the working practice, for example, including deliberations and decision making:

"I had previously considered the order of actions in my annotation procedure to be partially driven by the needs of the system ... When challenged about the order of the steps it became apparent that the annotation routine displayed a logical progression and was not an ad hoc series of steps".

The SKI system itself provided a means to formalise and encapsulate knowledge of which the expert was already aware. Using end user knowledge manipulation techniques enabled a direct and immediately accessible methods for testing existing knowledge.

"... subsequent to these results, I would ascribe a very low cue value to this measurement because it can only be used when a large number of assumptions have been made about the prosodics of the utterance, plus the syllable and utterance position of the relevant segment and its voicing. Although stridency has always had a secondary value in my hierarchy of cues, I now consider it to be of even less value than I previously thought ..."

In addition to the knowledge externalisation, a significant new finding was that the system was also a key component of the externalisation of what could be termed 'implicit' knowledge. The externalisation of such knowledge involves understanding its structure and how it can be expressed, not just making something that exists already available: it is, in effect, new knowledge. For example, an awareness that a hierarchy of cues is employed when interpreting the speech signal emerged.

"The hierarchy is flexible and adapts according to the features which are present/absent and their relationships. This hierarchy is implicit but is gradually becoming more explicit as I try to incorporate this flexibility into the rule base."

The frequent assumption when carrying out knowledge acquisition that experts are characterized by having complete knowledge did not apply in this case. The nature of scientific investigations and the use of innovative methods (i.e. the SKI system) pre-supposed a continually changing set of ideas and resulted in new perceptions. The process involved
continually re-assessing the immediate tasks in the light of experience and relating the results to longer term strategies. For example, when considering whether rules developed using the existing system will be re-useable and the impact of the technology:

"It is possible to use strategies which are suitable for the present research interest but which would be incompatible looking at these events later. An awareness of these events a rethinking of the strategy - so it can be incorporated into another rule base instead of using something which applies only to this instance - it has to be flexible enough to be re-usable as a basis for another rule base which can take this sort of event. It raises the problem of designing a rule base with hierarchies - the need for this is becoming more and more apparent. This leads to questions and solutions which are bound by current technology."

During both the spectrogram annotation and knowledge base development, new insights beyond the immediate concerns arose. Strategies were considered for use in other contexts and new requirements from the system emerged. This has implications for future rule base construction and the use of knowledge at different points in the process and for the exercise of control over the rule base structure by the user.

New requirements arose during the use of the existing system: for example, statistical support in the system was identified as support to the evaluation of the rule base constituents, as differentiated from support to the analysis and formalisation of knowledge about the visual features of the spectrogram.

Throughout the investigatory process using SKI, the stimulus to hypothesis was evident and insights arose leading to the definition of new requirements. An important requirement, that the knowledge base should be able to tolerate partial information where this is a consequence of the poverty of the acoustic signal or of the dynamics of continuous speech was identified.

7.1.5.3 Knowledge Encapsulation

The knowledge encapsulation process taking place during knowledge base development is a complex one in which the expert, in carrying out the task, holds a number of issues in parallel,
in particular with regard to the content, structure and expression of rules. A number of these issues were identified as follows:-

- There was a problem of handling increasing complexity in the rule sets.
  "The only problem with the system at present the rule base is the creation of robust contextual rules. This will become more complicated because this work is relatively straight forward, e.g. contextual value of succeeding or preceding segment".

- The role of testing was critical to the validation of the rules.
  "The logic is difficult to work out and follow through: the only way of assessing effectiveness is to test and examine results (i.e not check at the point of rule). When the rule does not succeed it is necessary to work though a maze of rules to find out why it wasn't satisfied".

- The significance of immediate feedback was recognized.
  "The inability to test immediately meant it took a considerable time to recognise flaws in the rules. It would have been preferable to have been able to test at the time of creating the rule".

The expert's commentaries indicated that significant decisions lie behind certain actions which are not apparent to external observation. For example, one approach to refining and evaluating rules was to apply a new rule even where the consequences for performance of the knowledge base were uncertain. New rules were written and tested wherever possible in order to gain information about what was needed. This action was taken even where it was already apparent that the rule, or a new constituent to an existing rule, would be likely to require change later. Indeed, decisions were assisted as much by negative results as much as positive ones: rules that failed gave rise to more questions and the pursuit of alternative strategies.

Two aspects of the development in the expert's thinking are especially significant and are expressed in the following remarks:
"It will be necessary to have an interaction between several layers of information....This is related to another area - of having a hierarchy of information that can take the phonetics of what it is given and then add in those items that are not retrievable from what can be seen on the signal.

- Strategies are needed for the contextual events.
  Many things not thought to be predictable by rule, are predictable by rule. It will necessary to have very tight organization in the hierarchy of rules..."

In attempting to apply purely phonetic knowledge to the construction if rules about the visual events in the spectrogram, it was apparent that other domain knowledge ('higher level' knowledge) was needed to provide complete information to the knowledge base. The application of different knowledge sources (e.g. phonetic, linguistic, signal processing auditory processing etc) therefore implicates the structure of the knowledge base itself. This confirms the expert's emerging realization that changes in the interface to facilitate rule application across more than the current three segments of the spectrogram was necessary but nevertheless not a solution.

These implications for rule base structure and the use of knowledge at different points in the process are major issues. The current facility to augment, enhance and alter the rules restricts the level of control to the existing underlying knowledge base structure. The rules themselves can be deleted or changed but they cannot be applied differently. A significant result was the expert's change from wishing to express knowledge about the segments as separate instances to a requirement to develop strategies for controlling the inferencing. That requirement is one that arose from the investigative opportunities afforded by the SKI system. It has considerable implications for the representation of the knowledge to the user.

7.1.6 Towards a Framework for Design: Setting the Goals

The implications of the results of the study are relevant to design of interfaces to Knowledge Based Systems as well as to our understanding of the process of knowledge externalisation. The suggestions for a design framework given below do not cover all the findings of the study.
It is important to remember that the design goals are especially applicable to the design of knowledge based systems systems that enable user manipulation of knowledge.

The assessment and manipulation of the source data raised important issues in relation to the handling of visual information by the expert.

It is important to provide methods for viewing all of the data at any one time and for viewing it in whatever form or forms are desired. In this case the data is very complex and the simultaneous availability of alternative ways of representing is invaluable. Graphical user interfaces that provide multiple windows on appropriately sized screens are the minimum requirement. The interface design should enable the user to tailor the display type and form at will and not constrain the number of windows available.

- **Design Goal**: enable an holistic view of visual data that includes multiple views

The relationship between being able to observe the source data and apply, text and amend the rules under development was important and felt to be a necessary facility. In fact, multiple views of the source data in combination with the rule creation tables were typically displayed up to seven windows at a time without problems. Graphical user interfaces that provide multiple windows on appropriately sized screens are the minimum requirement. The interface design should enable the user to tailor the display type and form at will and not constrain the number of windows available.

- **Design Goal**: enable simultaneous access to all forms of relevant source data and rule base input methods

The order in which specific decisions were made was not consistent. A number of factors were at play here. One was that a logical partial ordering, a form of hierarchy of cues, was not explicitly understood at first and only emerged during practice. Another was a clear need to be tentative and to be free to reserve judgement on specific decisions. Closely related to this point were unplanned alterations in what was being done arising from an exploratory approach in
which options were kept open. This implies that expert should not be constrained to make
decisions before moving to another point in the interaction. It should be possible to retrace
decision paths and amend previous statement. There was a need for suspending judgement and
keeping options open during the exploration of new ideas.

- **Design Goal**: Enable flexible and exploratory mode of interaction

The physical marking of the source data using graphical techniques had an effect on how it
was perceived. This link between action and perception is, on the face of it, extremely
interesting and deserves a more detailed analysis and study.

- **Design Goal**: screen design for graphical interaction should take account of the impact of
  the method of interaction upon the user's perception of the visual data.

There are implications from the work concerning existing scientific knowledge. It was evident
that the work called some existing theories into question in that domain. It would seem that
there may be a significant role to play in scientific investigation for knowledge based systems
used in the manner of this study and a real possibility for providing support to innovative
work.

It was clear that the encouragement to reflect on practice, enabled the expert to develop greater
awareness of the underlying processes at work in the application of specialist knowledge to
the task in hand, in particular, that there may be personal implicit 'theories of action' being
applied as distinct from generally received theoretical knowledge.

Domain specific knowledge, of which the scientist was initially unaware, was employed in
decision making. An important outcome of this observation was that, as an awareness of the
use of such 'higher level' knowledge' arose, it was possible to put that fact to constructive use,
i.e. it enabled the expression of new understandings of domain knowledge. That the generation
of new knowledge occurs during knowledge externalisation using EUKMS is a key aspect.
In attempting to apply purely phonetic knowledge to the construction of rules about the visual events in the spectrogram, it was apparent that other domain knowledge ('higher level' knowledge) was needed to provide complete information to the knowledge base. The application of different knowledge sources (e.g. phonetic, linguistic, signal processing auditory processing etc) therefore implicates the structure of the knowledge base itself. This confirms the expert's emerging realization that changes in the interface to facilitate rule application across more than the current three segments of the spectrogram was necessary but nevertheless not a solution.

These implications for rule base structure and the use of knowledge at different points in the process are major issues. The current facility to augment, enhance and alter the rules restricts the level of control to the existing underlying knowledge base structure. The rules themselves can be deleted or changed but they cannot be applied differently. A significant result was the expert's change from wishing to express knowledge about the segments as separate instances to a requirement to develop strategies for controlling the inferencing. That requirement is one that arose from the investigative opportunities afforded by the SKI system. It has considerable implications for the representation of the knowledge to the user.

- Design Goal: the structure of knowledge base should be flexible to allow reflection upon the knowledge sources being applied and their relative importance.

Knowledge encapsulation using the SKI method proved to be more complex than anticipated. This was not because the SKI method in itself was difficult, indeed its very ease of use prompted extensive application beyond the initial tasks for which it was designed. Rapid feedback about the consequences of changing, deleting or adding a rule was very important to the development of the expert's thinking processes. Interaction with the system stimulated new hypotheses about the data, its interpretation and the appropriate scientific theories. The provision of rapid feedback about the appropriateness or otherwise of rules created or amended by the expert was critical to the evaluation process but consisted only of success or failure in identification of single phonemes. In the first SKI version, explanation facilities were not included because the typical trace facility did not provide useful information for the expert.
After further evaluation of the system, 'explanations' that displayed the events satisfied and not satisfied through a Why? and Why not? facility were incorporated and used extensively.

- Design Goal: enable rapid feedback to user about rules applied in knowledge base and give support to evaluation that includes negative and positive results

Implications for the structure of the knowledge base, as well as its content, arose during the study. A particular requirement identified was that the knowledge base should be able to tolerate partial information in any respect that incomplete or poor quality data might demand.

- Design Goal: enable knowledge base to handle (tolerate) partial information in any aspect of the system.

**Summary**

**Design Goals**

- Enable an holistic view of visual data that includes multiple views
- Enable simultaneous access to all forms of relevant source data and rule base input methods
- Enable flexible and exploratory mode of interaction
- Screen design for graphical interaction should include the impact of the method of interaction upon the user's perception of the visual data.
- The structure of knowledge base should be flexible to allow reflection upon the knowledge sources being applied and their relative importance.
- Enable rapid feedback to user about rules applied in knowledge base and give support to evaluation that includes negative and positive results.
- Enable knowledge base to handle partial information in any aspect of the system.
7.1.7 Conclusions

Lamel (1988) considered that the attempt to formalise rules for plosive identification may be premature until there is a better cognitive model of spectrogram reading. This view is oriented towards the idea that another person i.e. an knowledge engineer can develop a system. A knowledge based system such as SKI places a significant degree of control in the hands of domain experts. The approach described gave direct control in the development and encapsulation of the knowledge. In this way, the need for a complete conceptual model of the domain in advance of the detailed acquisition of knowledge was not assumed. Indeed, the notion that such a model can be achieved realistically was thought to be impractical given the nature of the expert's research. One of the most important aspects is not only the ability to encapsulate and formalise knowledge but the stimulus it provides to the domain expert in respect of the heuristics: for example, what is the higher-level and lower-level knowledge which is brought to spectrogram interpretation? Thus can only be discovered when a piece of information thought to complete is seen to fail, thus prompting introspection and investigation as to why. It is this process which will lead to a better model of the knowledge processes involved.

The development of knowledge takes place in the interaction between the expert and his or her environment. Within this, technological tools provide a method for knowledge capture, a means of evaluating and validating that knowledge and a stimulus to the development of new insights. Thus, a knowledge based system offers a special kind interaction where the knowledge is captured directly without the mediacy of a knowledge engineer.

The SKI system enabled a domain expert to externalise and capture knowledge about the phonetics domain in a form which was machine readable and, therefore, able to be verified. The method for expressing knowledge to the system took the form of rules applied to a representation of the spectrogram source data which also characterised the knowledge base representation. The method proved highly successful and led to more ambitious experiments using more complex data by the expert. From this approach to the encapsulation of domain expertise and the externalisation of expert knowledge, new understandings about the nature of the knowledge being applied and a broad framework for design goals to be established was
A number of EUKMS design goals have been identified and it is postulated that they have general applicability.

Acknowledgements
This work was partly funded by the U.K. Government Alvey Programme, Project MMI/069, 'Human Computer Co-operation'. Particular thanks are due to E. McDaid and T. Bayley for their work on the SKI system and Speech tools and to Nick Rousseau for comments on an earlier draft of this paper.

References


8.1 ARTEFACTS AND THE DESIGNER’S PROCESS: IMPLICATIONS FOR COMPUTER SUPPORT TO DESIGN


A study of the design of the Olympic LotusSport Pursuit bicycle has provided valuable insight into the process of moving from a traditional, familiar design artefact towards its reformulation in both form and structure. The paper illustrates how new ideas developed from existing models and how conventions were used, changed and reformulated until an innovative concept arose. The history of the transformation of the bicycle artefact in relation to its predecessors is described. The total set of design instances provides evidence about the progression of the designer's knowledge from the initial learning of craft skills to expert knowledge applied in ground breaking design. From this evidence, a model of an innovative designer's process is derived and related to some existing models of design. The distinction between organisational models of design and the individual case is made. The paper concludes with some implications for computer support to the early stages of design, in particular the concept of a knowledge support system is proposed and illustrated.

Keywords: innovation, conceptual design, design artefact, design process, computer support, knowledge support systems

8.1.1 Introduction

Innovation in design is a key to competitive edge. The carbon fibre Monocoque bicycle ridden by Chris Boardman in the 1992 Olympic games is a recent example. That combination won a gold medal but subsequently the relative allocation between rider and machine in performance gain became contentious. Nevertheless, Mike Burrows, the original designer, has earned a place in the history of design engineering as a result. There is no doubt that the design marks a major conceptual shift in the structural, visual and aerodynamic characteristics of a bicycle. According to Richard Grant, it is "the most significant advance since the invention of the Rover safety bicycle." [GRA92].

---

1 A Monocoque is defined as a completely closed thin wall unitary load bearing-shell construction which cannot be analysed as individual load-bearing members. (Lotus Engineering).
A study of an innovative artefact and the designer's process, from which this paper is drawn, has provided evidence about the process of moving from a traditional, familiar design artefact towards its reformulation in form and structure [CAN94]. This paper illustrates how ideas developed from existing models, how the conventions were learnt and changed and how an innovative idea arose and was carried forward. The history of the transformation of the bicycle artefact in relation to its predecessors is described.

The total set of design instances provides evidence about the progression of the designer's knowledge from the initial learning of craft skills to highly expert knowledge applied in ground breaking design. The paper concludes with some implications for computer support to the early stages of design, in particular the concept of a knowledge support system is proposed and illustrated.

8.1.1.1 Innovative Design Process

From the evidence of the study, a model of an innovative designer's process was derived and related to some existing models of design. The distinction between organisational models of design and the individual designer's case is made. The innovative design process is as much one of identifying, defining or selecting problems as it is of solving problems. The designer generates a set of scenarios or possible prototypical solutions. A potential problem, in relation to innovation, is that designers tend to eliminate options early on in the process [LAW80]. Lewin makes a number of points about the process, in particular, the "extensive" nature of iteration and the importance of the specification-synthesis loop [LEW79]. It has become normal to emphasize three aspects of this process: analysis, synthesis and evaluation. These models represent the activities of groups of people in the task of design. The design process models that have informed the study and their relationship to the study findings are described in section 3 below.

8.1.1.2 Design Artefact and Designer Process

It is important, to distinguish between what is judged to be innovative by a consensus of views, based upon whatever criteria is appropriate (market share or success, performance) and the design activities leading to that result (the innovative design process). This distinction is analogous with Boden's separation between H-creative (historical) and P-creative (psychological) [BOD90]. Boden proposes two categories of creative ideas, concepts, artefacts and styles of thinking: those of historical creativity ("H-creative") and psychological creativity ("P-creative"). In the former case, the distinction applies to those
ideas that are novel with respect to the whole of human history, i.e. ideas that are first credited with originality. Psychological creativity, on the other hand, is such that it occurs within the individual mind, i.e. the person experiences the idea as fundamentally new whether or not others have had the same idea. That person may not be aware of other similar ideas or, indeed, may not recognize the significance of the idea at the time of its creation.

The approach adopted in the study reported was to try to understand the relationship between the outcome, the artefact, and the way it was designed, the designer's process. This was, in effect, to take an example of H-creativity and examine how it was designed from a P-creativity point of view. The results of that study may be compared with other studies in order to establish common ground with them and to propose general principles for the innovative design process.

The main point here is that in trying to understand and describe innovative design, it is important to make a clear distinction between trying to describe it in terms of a artefact and, by contrast, the means whereby it is achieved. In other words, we must not conflate the process of generating new ideas with the structure and composition of the artefact itself. Studying the design artefact alone in order to understand the creative process implies a kind of 'mapping' between the thoughts and actions and their outcomes. The case for assuming such close contingency is questionable [CAN92].

As a part of the retrospective investigation of the conceptual design of the LotusSport bicycle, the human designer's process in achieving an innovative design was investigated. The complete history of earlier bicycle designs that preceded the innovative breakthrough was studied [CAN94]. However, this is a partial record only and further study was made of the personal background, cognitive style and the context of the work. This result is a description which, whilst incorporating specific instances of designs, i.e. artefacts or products, may apply to the totality of the designer's activities. The outcomes of such a process may be innovative (or not). In the specific case under consideration, the carbon fibre monocoque racing bicycle was acknowledged to be an innovative, indeed, revolutionary design.
8.1.2 Design Process and Design Artefacts: A Case Study

When Mike Burrows began his bicycle design work, he was a novice with strong motivation, some previous engineering experience and an ability to learn quickly. The emergence of a principled design approach took place over time as the scope of his activities extended. As the history demonstrates, there were many factors that contributed to the final successful design product, not least of which was the knowledge gained from his parallel design thinking.

The forerunner bicycle artefacts that gave rise to the innovative carbon fibre monocoque provide a history of design process from which we can learn much about the way the designer's knowledge evolved. That process is the main focus of this paper and is described in 2.2 below. However, the artefacts by themselves do not provide the full picture of the innovative design process and, it is necessary to consider the cognitive issues as well. Those issues are described in full in [CAN94]. However, a summary is provided in 2.1 below in order to indicate the wider scope of the study and to illustrate additional sources for the requirements for computer support systems.

8.1.2.1 Design Process and Cognitive Issues

Design as practised by Mike Burrows is a diverse set of activities in which the interplay between many different factors is evident. Much of what goes on also can be found in routine design but it is the combination of being able to set high level goals and a strategy of taking the ideas from concept to artefact and testing in real use, that characterises his approach. In our example, all stages of the design and building of the bicycles were carried out by the designer from the initial generation of ideas, the rough sketching, tube brazing and component making in steel and aluminium to the testing in competition. The cognitive factors, in combination with the evidence about design and knowledge drawn from the artefact history have implications for computer support. Some examples are as follows:

- There is a total immersion in generating ideas. The majority of his ideas arise from pure mental activity: however, the roles of rough sketching and the made object and in developing, refining and transforming those ideas are significant.
Adopting an holistic or systems perspective on the design process is necessary in order to reveal the full scope of the problem. In order to achieve this identifying the unanswered questions was needed. The questions themselves provided a systematic decomposition of the problem at a high level.

The space of his design was not confined to one product area. It was the ability to make analogies with objects, machines and products in other areas that enabled him to transform the design space.

The immediate realisation of an idea in a concrete form, often without any intermediate drawings or just a rough sketch is a characteristic of the overall strategy. Designing "between my ears" and drawing on paper did not provide sufficient feedback: it was the thinking "with my hands" that was essential.

The methods used arose directly from the designer's overall strategies. What is significant from a design process model perspective is that the methods were acquired in response to need. Thus, the strategies drove design not the methods.

Sketching provides a means to explore the visual relationships of the elements of the design under consideration. Using a model, the "ready made" object is a common starting point. Sketching has a limited role in the eyes of this designer because it does not take him into the detail of engineering the object.

Being steeped in the historical and practical considerations of whichever field was being addressed at the time enabled the designer to draw upon significant domain knowledge. He constantly keeps abreast of new events in the field. The attention is to what other people are doing. The main point of the communication and reading is to derive design ideas and solutions for current problems in hand.

The above represent a sample of the issues that arose from a study of the designer's cognitive processes. In the following section, we focus upon the outcomes of that process, the artefacts themselves, and what is revealed by their developmental history in respect of the designer's domain knowledge evolution.

8.2.2.2 Design Process and Bicycle Artefacts

The design of the revolutionary single unit carbon fibre bicycle arose from a complex process in which the designer moved from adapting existing models and customising them
to suit individual requirements, towards a complete re-formulation of the guiding principle of the design of the bicycle i.e. to maximise the aerodynamics. By the time the monocoque frame emerged, there had been a radical transformation in the designer's process and the knowledge he had acquired and was applying. As the bicycle designs appeared, they were implemented and tested in competition and changes took place towards a more principled and analytical approach to his designing. Moreover, his knowledge, not only about bicycle design, but about practical aerodynamics, materials processing and engineering techniques advanced considerably.

The advancement of the designer's knowledge on a broad front took place during the process of applying his ideas from concept to realization. This began by learning to make bicycles out of manufacturers' tube sets and involved controlling all stages of the process. As the transformation from steel tube diamond frame to single unit carbon fibre monocoque took place, so did the nature of the design and development process. The scope of the activities was extended and necessitated a collaborative team effort. This was as a result of introducing radically different methods, especially in respect of materials processing, into the scope of the activities.

The Burrows' bicycle designs range from the early playful "Funny bikes" to the Windcheetah carbon fibre monocoque bicycle that was refined by Lotus Engineering into the Olympic LotusSport Pursuit bike (Figure 1). They embody experimentation and innovation in design that span a period of ten years. The development of the designer's knowledge associated with the bicycle design artefacts and the particular features of the design process are described below.

**First Bicycles**  
*Adopt, Adapt, Improve*  
*Learning Conventions*

The first bicycles were designed using a method inherited from early aero-modelling experiences where to adopt, adapt and improve was the usual approach. These bicycles were built without any application of design principles and the opportunities for creative design were minimal. Learning the conventions of the field was the main outcome of that period. That involved developing the craft skills of brazing and mitring tubular frames and selecting and fitting the components such as gears, brakes etc. Later, when he began to customise these traditional racing bicycles to match other cyclist's individual; requirements,
Figure 1. Olympic LotusSport Pursuit Bicycle ridden by Boardman 1982

Figure 2. Universal Frame 1982
the first steps towards developing the techniques for optimising the weight and handling features were made.

**Funny Bicycle Exploration Break Rules**

The 'Funny' bike was an exercise in exploratory design in which the existing rules for bicycle design were broken. It was based upon a conventional diamond frame structure but was an unconventional interpretation. The idea was to take the existing conventions and to break the "rules" and see what happened. By taking one element, the top tube, and sloping it forward as far as was thought practicable, the effect was to push to extremes (distort) the existing angles of the frame such as the head tube angle and seat tube angle. The result was a successful bicycle which was ridden in local competitions to good effect.

Here, the outcome was an innovative design but one achieved without any guiding principles. This stage was, however, a critically important one in laying the ground for moving away from existing models. This exploratory and, to an extent, serendipitous style of working established the basis for breaking from convention later on. Without this phase, it is questionable as to whether he would have eventually broken out of the conventional design space. In retrospect, Burrows acknowledges the importance of this period when he learnt how to explore a concept without worrying too much about the result, whether in terms of feasibility or performance.

At the time, the 'Funny' bicycle in itself did not represent a serious desire to create a truly original concept. It was the next generation that marked a turning point. The Human Powered Vehicle (HPV) design that was going on in parallel from 1980 enabled him to adopt a different perspective on how to approach design. This form of design involved creating new solutions in a domain without constraints which proved to be an ideal test bed for experimental ideas.

**Universal Frame Analysis Formulate Problem**

The Universal frame was a transition stage between the exploratory design of the 'Funny' bike and the innovative monocoque frame (Figure 2). By placing the principle of *maximising aerodynamic* advantage first, the total design was driven by that principle and led to significant consequences. Thus, a change in priorities gave rise to a major shift in design thinking.
The bicycle's frame was reduced from 23 inches to 16 inches with a flat top tube and an extended seat pin to accommodate rider size. The result was a new design that had been evolved from the previous model but kept the basic form, structure and material of the original. The transformation in perception of what a frame could be was realized at this point because the traditional focus on frame size and angles was dispensed with and the aerodynamic properties of the frame and other components took priority. The influence of a parallel design activity, that of Human-Powered Vehicles (HPVs) had enabled him to break out of the conventional cyclist's way of thinking about the frame and all the relevant components and to envisage it as a whole unit.

The significance of this bike was critical because it inspired the monocoque frame. The frame was so small it became possible to visualize it later in a different way that was to lead to the idea of filling it in to improve the aerodynamics. Thus, the problem was ready for a solution and when carbon fibre materials appeared the opportunity was seized.

**Monocoque Mark I  Emergence  Evolve New Concept**

With the single unit Monocoque frame, the fundamental character of the bicycle itself is changed. Not only is the appearance of the diamond frame dispensed with, but the underlying structure has been totally transformed: it is a one piece carbon fibre unit. Such a form may be described as *emergent* in that it displays none of the explicit characteristics of its original source (Figure 3). Thus, the perception of the experts in the field is that it was not a true bicycle and indeed the bicycle was banned in 1986 by the UCI, the International Cycling Federation. For a discussion of emergence and its implications for the design of computer systems see [EDM94].

The primary "philosophical" goal of the designer was to mould the frame as one piece. This proved to be very difficult to realize but nevertheless he persisted with it. The visualization of the frame was done by sketching over pictures of existing conventional bicycle frames (Figure 4).

The factor of the single body made as a unit in carbon fibre meant that all the traditional measures for coping with stress on the bicycle were removed. The components of the bike in combination were made to support the principle of aerodynamic efficiency. To that end, the frame contravened the normal rules and expectations for a racing bike.
PAGE NUMBERING AS ORIGINAL
The decision to use carbon fibre material for the monocoque frame was an "opportunistic" strategy. The problems inherent in making with carbon fibre rather than steel tubing were yet to be discovered. Once that idea was conceived, however, it then became necessary to devise new processes for making the frame. Shaping and moulding carbon fibre is closer to dressmaking than traditional engineering.

There was no previous experience to draw upon and it became necessary to gain access to the expertise of others. This had an effect on his design practice because he had to produce an accurate drawing with dimensions in order that his ideas could be converted into the master mould. Prior to that, preparing a specification was not part of his normal design practice, nor indeed was working with a team, albeit very small. The materials processing itself was a difficult one which involved considerable improvisation. Here, the learning curve was steep but the resulting knowledge was to prove significant for future work.

**Monocoque Mark 2 Analogy Modify Concept**

This bicycle had the same specification of the Monocque mark I with one exception that was to prove an enduring innovation, the addition of the single front fork, the "Monoblade". The origin of this idea came from deep in bicycle history. Making the connection with the Invincible Bicycle design of 1898, seen in the Coventry Museum, and recognizing the potential of the single front fork was a sign of deep immersion in the knowledge of the domain. Thus, *analogical thinking* was to play an important part in advancing the design.

That fact alone does not explain the way the idea was formulated. It was being able to recognise another instance of turbulence cause by twin forks, (that of the bi-plane) and the implications for streamlining that it implied. Thus, being able to cross reference between different product areas is an important advantage to design innovation because it supports the breaking out from a single conceptual design space. The transfer of ideas take places as a result of seeking analogies in associated designs.

**Inter Bike Refinement Add Features**

This bicycle represents a *consolidation* phase rather than a highly innovative one. The additions made were the result of checking out ideas that were already implied, having been brought about by the knock on effects of the monocoque frame for other components (i.e.
wheel size). The small front wheel had no longer been necessary from the realisation of the monocoque frame but it was felt necessary to test the impact of change on the handling very thoroughly first before committing to the larger wheel. It is important to recognize that not all conceptual leaps happen at once and that the implications of some design decisions for others are not necessarily apparent at the outset.

**Monocoque Mark 3 Synthesis Combine Features**

The third Monocoque bicycle was to be the one that incorporated the best design innovations of the previous bikes: from the extra small 16 inch universal frame bike, through to the breakthrough carbon fibre monocoque frame version and its successor with the monoblade. The principle of achieving the frame as a single unit was applied. As an embodiment of all the successful design features it was the result of a truly holistic design endeavour. The frame design modification and improvements in the frame moulding process were made possible by much hard earned experience. The bicycle represents the culmination of ten year's growth of design knowledge.

**Olympic Bicycle Completion Apply Measures**

The basic specification of the first version of the LotusSport Pursuit Bicycle was the same as Mike Burrows' Monocoque mark 3. The introduction of more rigorous testing using the MIRA wind tunnel lead to modifications in the frame profile and other features. These changes differentiate the Burrows' bicycle from the Olympic bicycle.

The first tests with the rider in position revealed little advantage over his own conventional racing bicycle. The main change was to lower Boardman's position even further and because he could maintain that, the turbulence was reduced even further. An account of the development of this bicycle is to be found in [HIL93].

The opportunity to exploit the Burrows' Monocoque design was grasped by a company with an entirely different product profile to the cycling industry. The effect of Lotus Engineering's corporate code of style, performance and charisma was to envisage the bicycle in that frame of reference. Once the idea had been taken up in the company, its survival towards the Olympic bid depended upon a number of champions. There are lessons for organizations about discovering and fostering innovation in this experience.
8.1.2.3 Knowledge Development and Design Process

The previous section described the individual bicycle artefacts and what was revealed about the evolution of the self-taught designer's knowledge over that period of time. Figure 5 gives an overview of the bike designs, design process characteristics and the knowledge evolution that took place over more than ten years.

Knowledge accrues gradually, beginning with copying and adapting designs as part of the initial learning of conventions process. Once the rules and skills-base for the craft is established, then some customisation takes place, but still adhering closely to the existing models.

In an exploratory phase, the deliberate breaking of conventions signifies a desire to try out ideas to see what will happen, a kind of serendipity where the results are not predictable. The impact on design practice is to free up the designer from thinking in a consistently convergent manner. Designing in this way only gives way to a more principled approach when the design problem is formulated at a higher level. In this case, it occurred when the designer analysed the problem as one about aerodynamics of the whole bicycle not just the improvement of weight and handling that existing design approaches addressed.

<table>
<thead>
<tr>
<th>Date</th>
<th>Artefacts</th>
<th>Design Process</th>
<th>Knowledge Evolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1979</td>
<td>FIRST BIKES</td>
<td>ADOPT ADAPT IMPROVE</td>
<td>LEARNING CONVENTIONS</td>
</tr>
<tr>
<td>1980</td>
<td>FUNNY BIKES</td>
<td>EXPLORATION</td>
<td>BREAK RULES</td>
</tr>
<tr>
<td>1982</td>
<td>UNIVERSAL BIKE</td>
<td>ANALYSIS</td>
<td>FORMULATE PROBLEM</td>
</tr>
<tr>
<td>1985</td>
<td>MONOCOQUE 1</td>
<td>EMERGENCE</td>
<td>EVOLVE NEW CONCEPT</td>
</tr>
<tr>
<td>1986</td>
<td>MONOCOQUE 2</td>
<td>ANALOGY</td>
<td>MODIFY CONCEPT</td>
</tr>
<tr>
<td>1988</td>
<td>INTER BIKE</td>
<td>REFINEMENT</td>
<td>ADD FEATURES</td>
</tr>
<tr>
<td>1990</td>
<td>MONOCOQUE 3</td>
<td>SYNTHESIS</td>
<td>COMBINE FEATURES</td>
</tr>
<tr>
<td>1992</td>
<td>OLYMPIC BIKE</td>
<td>COMPLETION</td>
<td>APPLY MEASURES</td>
</tr>
</tbody>
</table>

Figure 5. Bicycle History Design Process and Knowledge Development

From the problem formulation point onwards, the design artefact is viewed differently and the problem tackled as an holistic one where each part of the artefact is subjected to the overriding principle of designing to maximise aerodynamic advantage. As the level of concern lifts, so do the possibilities for innovative solutions.
When the Universal bicycle was conceived, the small size of the frame made it possible to envisage it as one unit but, until the advent of carbon fibre, that was not feasible. The preparedness for seizing an opportunity and exploit the new material when it arrived was already established. Such was the designer's expert state of knowledge by this time that he was able to recognise the implications of the new material. Nevertheless, without the strong motivation and high level goal of achieving a holistic design, he might not have persisted with the challenge that processing the carbon fibre posed.

With the design and making of the Monocoque bicycle, it was at this point that the individual designer became a team player. With the advent of the need to create new processes for making the artefact, came the requirement for the expertise of other people. The individual designer had become a team member, albeit in a leading role.

A number of issues can be drawn from the above discussion that must be taken into account when developing computer support systems for design. During the design process the designer is constantly learning and gathering knowledge from a wide range of sources. In relation to which, we see the evolution of concepts and the occasional emergence of new, innovative, concepts. This evolution does not necessarily take place within the closed set of rules that represent conventional wisdom. Indeed, the creative step is often one in which a conventional rule is broken.

Perhaps the most important issue to note is that the same design process can be represented in a number of different ways, depending on the perspective taken. We briefly visit this point below, but can note now that implications for computer support must be drawn from the innovative individual's perspective as well as the organisation's. Computer system design must consider cognitive as well as organisational process issues and, for that concern, a study of the relevant artefacts will not suffice.

The realities of the design process are vital concerns when providing support. Idealised models lead us to idealised support which is only too likely to fail in practice. Different models can be constructed for different purposes and in the next section we provide an overview of some of the perspectives that exist.
8.1.3. Design Process Models

8.1.3.1 The Design Process Observed

Drawing upon the design process revealed by the study, one can represent this process as a spiral, in which the emphasis on the sector shifts as the design develops. Thus, in a certain sense, all activity types are ever present whilst the focus shifts as seen in figure 6. Looking at the spiral we can see, for example, that a generator might be selected in the 'make' stage, leading to a conjecture, or idea, in the next step, etc. Hence, by taking the spiral view, we can begin to see how the various models at their different levels of granularity may be brought together. For the purposes of this discussion, the spiral model is taken as an adequate picture of the process under investigation.

![Figure 6. A spiral view of the design process](image)

In certain respects, the spiral can be seen as a synthesis of other models of the design process. Below, we review some of those models as background to the considerations of the paper.

8.1.3.2 Existing Models of Design

The word Design has many different connotations. These range from an interpretation that goes no further than the superficial elements of a product added to make it more marketable (sometimes called aesthetics), right through to the notion of a complete continuum from
the initial formulation of the problem to be addressed (including marketing strategies), to the detailed specification.

An overall representation of the Design Process might include initial requirements, concept design, proof of concept (early prototypes), preliminary design, detailed design through to full prototypes and manufacturing. Another important consideration, from an organizational point of view, is the collaborative effort required. Design is both a team effort and an individual one.

A standard view of the design process is Lewin's [LEW79], which is represented in figure 7.

His "analysis of the solution" activity is often termed "evaluation" [LAW80], which indeed it is. Thus moving between the problem, synthesis and evaluation is seen as fundamental to the process. He makes a number of points about this model. In particular, the "extensive" nature of iteration and the importance of the specification-synthesis loop in relation to Research and Development and hence, perhaps, innovation.
This type of model represents design from a point of view that makes the organizational perspective clear. It is concerned with the information flow in an organisation, ignoring information that is used to facilitate the process and then discarded. In particular, it represents documentation and the flow of internal deliverables rather than the details of the individual designer's actions.

Looking at the issue from a more cognitive point of view, [DAR78] used empirical evidence to develop a variation which she termed generator-conjecture-analysis (Figure 8).

![Figure 8. Darke's view of design](image)

More recent empirical work has supported and extended Darke's view (LAW93). Lawson confirms the importance of the primary generator (an early concept or model used to form the design idea) and the use of governing principles that guide the design conjecture [EDM87] or shape the synthesis. Lawson's earlier model of the creative process [LAW80], as shown in figure 9, is not contradicted by his more recent work. However, the recent work is differentiated by his drawing out of the significance of the maintenance of "parallel lines of thought" by the individual designer.
The variation in these models of design can be understood best in terms of the different granularity's of concern of the models, from the organisational process to individual thought. In fact, if one looks in more detail at the processes within each stage of, for example, figure 7 one finds the whole process mirrored in a certain sense. For example, during the ideas stage a certain amount of synthesis, evaluation and making, no matter how tentative, may be observed [MCN94]. Looked at from the larger perspective, of-course, the early physical making, for example, does not figure highly. It tends to be used and discarded rather than passed on to the next stage. However, it does take place and must be taken to contribute, in some way, to the process.

Figure 9. Lawson's model of the creative process
8.1.4 Computer Support for the Designer

The designer is a knowledge worker involved in creative work that is not easily characterised by formal procedures. Computer support, therefore, is not most easily provided by implementing well defined processes. What is required is support that is flexible and usable in the context of the attributes of the thinking processes that the designer is engaged in. Our study has identified a number of the relevant issues and we can consider them in relation to the implied computer requirements.

- **Break with convention**
The ability to break away from conventional expectations, whether visual or mechanical, is a key characteristic of the innovative thinker. Direct interaction with knowledge in the system is a way forward here. This requirement places particular constraints on knowledge access, acquisition and validation.

- **Immersion**
Immersion in generating ideas for the design and engineering of machines is indicated by continually heightened mental awareness. This implies portable and all-pervasive computing.

- **Whole process**
Adopting an holistic or systems perspective on the design process is necessary because, only in this way, will the full scope of the problem be revealed. Thus the designer needs to be able to readily see overviews of the problems and the developing solution. Multiple views of the data also seem to be important.

- **Parallel channels**
Keeping a number of channels open in parallel, especially where the problem is a complex one, is a necessary part of generating design ideas. Many problems are solved by analogy. Thus the user must be able to multi-task and switch between tasks freely.

- **Sketching**
Rough sketching has a role in defining an idea already conceived as part of a longer process towards realising it as a physical object. Using a model, the "ready made" object is a
common starting point. Sketching also provides a means to explore the visual relationships of the elements of the design under consideration. Tentativeness is an important characteristic of sketching that must be supported.

- Gestation

The gestation period for the innovative ideas to be conceived, explored and developed was over a considerable period of time. During that time, the evolution of the artefacts took place in parallel with the acquisition and refinement of the designer's expertise and knowledge. Long term personal knowledge and databases are implied.

8.1.5 Discussion: Knowledge Support Systems for Design

A Knowledge Support System enables direct interaction between domain expert users and the knowledge in the computer system. This type of system is designed to address the problems of the ineffectiveness of indirect knowledge capture using the mediation of a knowledge engineer [SHA88; GA90]. Being able to interact with a knowledge-base enables the domain specialist to examine and evaluate that knowledge closely. In addition, having assimilated the results both the exploration of the visual data and the knowledge base, it is possible to advance the existing knowledge further.

The designer has individual design knowledge as well as the shared knowledge of design practice. Indeed, designers' success depends upon the distinguishing contributions that they bring, as is true in all professional practice. Hence successful support systems cannot only provide pre-canned knowledge but must tackle the problem of the acquisition of knowledge of the individual designers. Specific knowledge, of course, must also be acquired at the time of design.

In earlier work, a case for providing experts with direct access to knowledge-based tools in order to overcome some of the inherent problems was made [CAN93; CAN93]. The Knowledge-Support System will enable the designer to both externalise and capture knowledge without the help of a "knowledge engineer". This process is a dynamic one where the interaction between knowledge worker and knowledge-support system leads to new understandings. Thus, the knowledge of the domain expert evolves as new
understandings emerge. That evolving knowledge brings with it additional requirements for support.

It is necessary here to distinguish between something that is *learned* and something that is *generated*. In the latter case, from the designer's point of view, new insights may appear as unexpected revelations about existing knowledge and, in that sense are not learned in the traditional sense of the word.

The dynamics of this situation naturally lead to special demands of the design and development of the system. For the interaction between designer and knowledge-support system to be effective and productive, it must be smooth, harmonious and in accordance with the domain specific 'language' of design.

### 8.1.6 Conclusions

A Knowledge Support System is being developed at the LUTCHe Research Centre that addresses the issues identified above. The approach is to give the designer direct access to historical knowledge, the rules of received wisdom and databases of technical information. A key element of the approach is to enable the designer to review, evaluate and modify the knowledge in the system. It seems that this facility is vital if creative work is to be assisted by computer systems.

**Acknowledgements**

A number of people provided information and comment on this paper. We are indebted to the following for their support and co-operation: Mike Burrows, Lotus Engineering, David Patrick and Bryan Murray. The work was partly funded by the ACME Directorate of the SERC. In particular, we would like to thank Mike Burrows for permission to reproduce his photographs and sketches.

**References**


268


8.2 A STUDY OF THE CREATIVE DESIGN OF THE LOTUS BICYCLE: IMPLICATIONS FOR KNOWLEDGE SUPPORT SYSTEMS RESEARCH


This paper is concerned with identifying the research directions for future computer systems for creative design. In order to define the requirements for support to the designer, or design team, we need to understand more about the way new ideas arise and come to fruition in an innovative product. We report a case study of the design of the LotusSport bicycle by its originator, Mike Burrows. In the light of this study and its relationship to other research into the characteristics of creative designers, we consider the implications for the design of computer support systems, in particular the knowledge intensive aspects of design. Support for interactive knowledge-based design requires fluent interaction between designer and knowledge in a way that does not impede the creative process. We term such a computer support environment a Knowledge Support System. Future research directions concerning interaction with design knowledge are proposed.

Keywords: Creativity, cognition, bicycle design, knowledge support systems, interaction

8.2.1 Introduction

Requirements for future computer supported design environments may be derived by using prescriptive models of design or, alternatively, descriptive models of design drawn from empirical research. The goal of the prescriptive model is to improve the design process and the resulting artefacts, whilst the descriptive model incorporates evidence about what happens when designers design in the real world. In practice, the development of computer support environments may combine both prescriptive and descriptive models.

There is a widely held belief that current Computer Aided Design (CAD) does not support the whole design process, particularly the early conceptual stages. There is also a school of thought that holds that computer systems have considerable distance to travel before they are to performs the role of enhancer or amplifier of human creative design. In particular, the notion of co-operative interaction between computer and human has yet to realise its full potential. A view that the focus of research should be upon the
development of computer-based environments that support the complementary nature of human designing and computer generated design is growing\textsuperscript{7,8}.

In order to achieve such goals, more understanding is needed about how design takes place and, in particular, the characteristics of and the circumstances in which creative design occurs. Design, as an inherently creative and exploratory activity, is at the centre of the research reported in this paper. In order to define the requirements of computer support to the exploratory and creative aspects of design, we need to understand more about the way ideas arise and are realised in physical form. We can also draw upon both prescriptive and descriptive models of design, but it is important to recognise that these span a wide range of domains and have yet to be generalised across those domains. In this paper, we have drawn upon a number of empirical studies from our own work and that of other researchers in the field.

The particular subject of this paper is a case study in creative design and its relationship to identifying the requirements for computer support. The intention of the case study was not to model the whole design process but to identify the opportunities and constraints for computer support to creative design. From that framework, the goal was to determine the features of a design support system, and in particular, the knowledge intensive aspects of the system\textsuperscript{9}.

The Lotus Sport bicycle, ridden by Chris Boardman to a Gold medal for the Pursuit event in the 1992 Olympic Games, marked a major conceptual shift in the structural, visual and aerodynamic aspects of bicycle design.

\begin{figure}[h]
\centering
\includegraphics[width=0.5\textwidth]{lotus_bicycle.png}
\caption{Chris Boardman riding the Lotus Bicycle, Leicester 1992.}
\end{figure}
This was "the most significant advance since the invention of the Rover safety bicycle" according to cycling expert Richard Grant. The significance of this innovation in giving added performance to the rider is, in the cycling world, still contentious. At the time of writing, it is not possible to buy this bicycle in the mass market. As Ballantine observed, the transformation of a totally innovative idea into a commercial prospect is a challenge in itself, especially given the limited profit margins available in the cycling industry.

Basalla argues that theories of technological invention and change have either tended to dwell upon the artefact itself, using an economic or social explanation, or on the contributions of individual insights, using a psychological perspective. However, these analyses are usually considered in isolation from one another and therefore, provide only a partial explanation. His approach is to include in any analysis the impact of large scale innovations brought about by individuals as well as small scale cumulative changes over time. He argues that there is a role for studies of individuals in providing depth to our understanding of the design process as distinct from the results of broad brush studies.

"Sheer number of inventions do not guarantee that a major technological change will occur. The key is always the inventor's act of insight by which certain elements are chosen, combined in innovative ways and made to yield a solution." (12, p 24).

Individual cases may provide the source material for the kind of analysis advocated by Basalla, of which the study reported in this paper is an instance. We report a case study of the design process of Mike Burrows, the originator of the LotusSport bicycle design. The main focus is upon his creative cognition: this refers to the combined effect of a number of elements of the designer's processes that were critical in the generation of new ideas and in the making of artefacts. In the final section, we propose some research directions for the development of computer supported design systems taking account of human cognitive issues and the interaction with domain knowledge. This research draws upon the results of the study reported as well as previous studies conducted by the authors and other research activity in the area of computer support for creative design.

8.2.2 Research Framework and Methods

The framework for the study was derived from our existing work and other studies of innovative designers (e.g.,). In Maccoby's study of a number of high profile designers and engineers, a set of criteria was identified which forms a basis for the analysis.
described in sections 3 and 4 below. Whilst such creative people are by the level of their success exceptional, they nevertheless may offer some pointers as to how to identify innovative practice and the conditions wherein it develops. Thus, design studies undertaken in different domains provide insight into the designers' cognitive processes. The design process models that have informed the study are described in.

Design process models can take many forms depending upon their purpose and origin. For example, if the model is to help organise paper flow through a design company, it is likely to have a procedural character, that is, it follows an order or a pre-determined sequence. On the other hand, if one is trying to understand early conceptual design, where many different paths may be taken, and vary from individual to individual, the complexity of this (largely unpredictable) situation does not lend itself to a procedural model. This does not mean that strong generic characteristics of the process cannot be modelled. In particular, descriptions of the constraints that operate may be much more appropriate to a model that can be applied in design.

In a retrospective study, the problem of obtaining accurate recollections from the designer must be acknowledged. However, where direct observation of the designing in progress is not available, it is still possible to recover the history of events with a reliable degree of accuracy. We consider that the designer's verbal accounts and the design output (sketches, products etc.) provide a dual source of information that can be analysed using an external investigator's perspective. Our approach was to investigate both the designer's recollections and what was revealed by the various artefacts created during the history of the development of the final design. The unravelling of a number of predecessor bicycle designs proved to be an effective vehicle for establishing a dialogue about what actually happened in terms of the designer's personal practice. The materials and tools of early design, the sketches, drawings and prototypes provided a record of the design process activities around which discussion took place. Two types of interviews were carried out: informal discussions followed by structured interviews, all of which were tape-recorded and transcribed. The transcript data was checked with the designer for accuracy and follow up interviews carried out. The results of the analysis are described in section 3 below.
8.2.3 Creative Design: A Study of Practice

In this section, a selection of findings from the data are presented. Two forms of analysis of the interview data were made. First, the bicycle design history was examined in terms of the way each design represents a progression or extension in the knowledge that the designer used. Second, the design process was examined in terms of the various activities that comprised the designer's practice. It is this latter process perspective that is the subject of this paper.

The Burrows' bicycle designs embody an individual designer's experimentation and innovation spanning a period of ten years from 1980 to 1990. They range from the early "Funny bikes" to the Wincheetah bicycle that was refined by Lotus Engineering into the LotusSport Pursuit bike. During this time, the design of the frame was transformed from the conventional diamond shape into a monocoque (single unit closed-wall) frame made of carbon fibre material. This design was, at first, banned from international competition by the International Cycling Federation (UCI) because it was classified as "illegal", a ruling that was overturned only two years later in 1988.

The advent of the monocoque frame led to a reorientation of the whole concept of the bicycle. At the time it first appeared, the design contravened conventional expectations of what a bicycle should look like, even in this experimental age. Whilst the frame was dramatically different in a visual sense from a conventional diamond frame, the distinctive feature of the Burrows' bicycle's - its very essence - to use the designer's term, was that it was an integrated holistic design in which all the individual components contributed to the main goal of achieving maximum aerodynamic efficiency.

8.2.3.1 Creative Cognition in Innovative Design

The Burrows' design practice comprises cognitive processes and physical actions in which the interplay between many factors is evident. The core activities are to be found in normal routine design but there are some key factors which distinguish the particular approach. For example, demonstrating a deep immersion in the knowledge of the field and designing in parallel product areas proved critical to innovative thinking. Another important feature of the designer's creative cognition was his holistic design strategy whereby devising concepts, making artefacts and testing in real use were managed and carried out himself.
The elements of the Burrows' creative design process should be seen as interlocking and interdependent activities that were developed over a long period of time and ultimately led to the monocoque frame breakthrough. For the purposes of analysis, the elements of creative design were categorised as: Ideas Generation, Problem Formulation, Strategies, Methods and Expertise and are represented in Figure 2. A selection of the findings from the study follows, supported by quotations taken from the transcript data.

![Figure 2 The Elements of Creative Design](image)

- **Ideas generation**

  Conventional wisdom has it that truly novel ideas spring up without forewarning, often without precedents. This case illustrates that the pathway to innovation can be a long and, indeed, tortuous one. The first monocoque frame bicycle was registered as a design in 1982, but it was not actually built for real use until 1985 when it proved feasible to mould the frame in carbon fibre composite material. However, it took until 1992 before the final model was refined into the LotusSport bicycle and appeared upon the world stage at the 1992 Olympics. During those ten years, many ideas were shelved and brought out when the time was right or a chance opportunity brought them back into play.

  From this designer's point of view, there were no ideal conditions for generating new ideas. Whether sketching in front of the television or spending hours on a bicycle turning over ideas in his head, he was immersed in some aspect of designing. One important feature of
the exploratory design thinking was a need to keep a number of channels open in parallel. The practice was to generalise and transfer the lessons from one item to another (e.g. from aero-models and Human Powered Vehicles (HPVs) to the bicycle):

"It's not committing yourself immediately to any particular channel, particularly if it's a complex problem. There's always more than one way."

This parallel stream of designing enabled him to formulate his main objective for the bicycle: (i.e. how to improve its aerodynamics) because he was able to perceive it in a different frame of reference:

"All the time I have been building bicycles I have been building HPVs and racing bikes. You learn the strengths and weaknesses of the two sides."

Thus, whilst the starting points were the existing traditions and models, it was his ability to make analogies with parts and products in other areas that enabled him to extend and ultimately transform the design space. That approach is a key aspect of problem formulation described below.

- **Problem Formulation**

Problem formulation in the design process under consideration, is the systematic expression of the complete scope of a problem with all its inherent inter-dependencies. This is to be distinguished from devising solutions to a problem that has already been described and for which the implications for design are understood. According to this definition, it means that the designer chooses problems with minimal constraints, considers issues beyond the detail of existing product or artefact and identifies previously unanswered questions. In this aspect of creative cognition, the designer finds ways of breaking out of conventional expectations. On the constraints of cycling traditions he notes:

"You want to make a better bicycle and therefore it has to be lighter, shorter, stiffer.... That is how cyclists perceive the bicycle. They were the guide-lines that tradition suggested ...and that I followed.... I used... sort of radical approaches.....leap frogging. But it was only after say a couple of years working with bicycles and HPVs that I really was able to totally free myself from conventional thinking."

In the Burrows' approach to problem formulation, the space of his designs is not confined to that of the bicycle. The attraction of HPV and aero-modelling was the lack of constraints which provided him with ample opportunity for devising new approaches. In this way, the
pre-conditions for being able to break out of the existing design space were established and continued throughout the important design years leading to the monocoque bike. The actual point of departure was when he identified the key question to be addressed, that of how to improve the aerodynamics. At the time, this approach was dramatically different from the normal practice of refining the existing frame attributes by reducing weight and tightening angles. He then understood the main constraints upon bicycle performance and the reasons that the problem had not been solved before.

"You just had to sit down and again adopt this holistic approach. Bring it all together from nowhere, no starting point. You only knew what the goal was....how fast can you go. You knew you solved that by making it more streamlined. To be very fast, therefore you have to be aerodynamic."

Taking a higher level view of the problem space proved to be a critical change that enabled him to step outside the existing constraints and dare to be radically different. Such problem formulation operated within the commercial strategy of a successful self-employed business man, that of identifying customer requirements, designing and building products and delivering them to customers. This approach was an established part of all the design activities that he undertook. The relationship of these issues to strategic thinking is described further below.

- **Strategies**

In this context we define *strategies* as decisions and measures taken in order to achieve goals. Methods, on the other hand, are sets of actions that carry out the strategy. Whilst strategies usually arise from considerable experience and are peculiar to the individual concerned, methods are more available to use by others i.e. they may be taught or be unambiguously described.

A characteristic of the Burrows' design strategy was the "design, make, use (test) and iterate" approach. Adopting such a strategy involved generating the initial ideas, taking those concepts forward as sketches or models, engineering a fully functional object and then testing it himself in a realistic way (e.g. by riding it in a race). The resulting feedback was used to improve the design further. The act of designing and making an artefact was necessary to a full understanding of what had been done. Designing "between my ears" and drawing on paper did not provide sufficient feedback: it was the thinking "with my hands" that was essential.

277
"I literally think with my hands. I very seldom draw any sort of dimensions on a piece of paper. I occasionally doodle things to work at them, but I'll basically just pick pieces of metal out of the rack and drill holes in them literally and it will get bolted together... Now either you've got to go back and redefine it or occasionally go laterally and then come out with an even better idea".

A change in strategy that proved significant was the decision to use carbon fibre composite for the monocoque frame. In order to specify the design, he departed from his existing strategy that of moving directly from idea into making the artefact. It now became necessary to draw the design more carefully and to specify dimensions in order that someone else who had carbon fibre materials processing know-how could carry it out. Thus, the move to a different material altered his strategy of controlling and implementing the total process himself to a more team-oriented approach.

A creative strategy may also involve risk taking that can lead to significant gains ("leapfrogging") or, by contrast, failures ("landing in the stinging nettles").

"Life is full of opportunities. You make mistakes...they force you into a new direction... every once in a while that gives a, "wow!" and idea you would never have seen."

Thus, having made mistakes, the response was to accept them for the impetus they provided to a change of direction. Opportunism as a strategy for change requires being receptive to new ideas and being able to adapt or devise new methods quickly to meet new circumstances.

- **Methods**

Methods are sets of actions that are carried out within a strategic framework. Making design ideas into working products required the necessary methods and, therefore, craft skills play a significant part. In bicycle design, the customisation for individual requirements is characteristic of the competitive cycling community. Burrows learnt his craft skills because he needed to realise some design ideas that could not be commissioned elsewhere. However, he had no interest in craft for its own sake. His usual practice prior to the move to carbon fibre was to make parts in aluminium or steel using conventional lathes and milling machines. These methods involved applying engineering skills to carry out the general strategy for testing out design ideas in a concrete form. The origins of what might be termed the "adopt, adapt, improve" method was in early aero-modelling days: for example,
there'd be an existing design, maybe you built it as a kit or seen it as a plan and you would modify it, and that's your starting point to stray away from the existing ready made designs and you modify that until you end up with your own design.

In Figure 3, the sketching over an existing diamond frame shows an intermediary stage moving towards the full monocoque design.

Figure 3 Monocoque sketch over diamond frame

Taking up a new idea or material and deliberately pushing the boundaries of what can be achieved can result from an opportunistic strategy. A distinctive feature of the Burrows' approach was a nose for adventure and risk that led him to seize upon new ideas for pure experimentation's sake. The moulding of carbon fibre for the monocoque frame was such an instance where he embarked upon the exploitation of this material without being inhibited by a lack of prior experience. The methods for moulding were learnt by dint of experience and the failures were considerable. What is really significant from a creative design point of view is that the methods (skills) were acquired only when the need arose.

In this way, the strategy drove the design process rather than the methods themselves. The methods were developed or acquired as a result of the change of strategy. Thus, the limitations that manufacturing issues might have imposed upon the design in the making the monocoque frame were not the driving force. Had he considered in advance that the
moulding of carbon fibre composite would prove to be so difficult, the design might never have been realised. In this case, pushing the frontiers forward was not inhibited by the implementation constraints.

- **Expertise**

Burrows' domain expertise was acquired in response to his particular needs rather than being shaped by a formal education. However, this did not mean it was driven by short term considerations. Adopting a more principled approach to design did emerge during the course of his self-taught endeavours. Being immersed in the historical and practical considerations of whichever field was being addressed at the time enabled him to draw upon significant domain knowledge.18

In addition to his existing knowledge and expertise, he constantly kept abreast of new events in the field. This was a form of practically oriented "research" that was characterised by the following:
- he kept in close touch with what other key players were doing
- he had a network of contacts with experts active in the field
- he read the technical literature for ideas being applied in practice

In the following section, the individual case is placed within the context of other studies of creative work.

**8.2.4 Creative Design : Discussion**

Approaches to the study of creativity and original thinking are based upon the different goals and methods of disciplines, ranging from Cognitive Psychology to Artificial Intelligence e.g.23,24,25. Studies of cognition focus mainly upon the internal processes of the human mind26 whilst others argue that so-called intuitive "hands-on" experience, i.e. our perceptual experience, is essential for the interpretation of our mental concepts27,28. There is a considerable literature about the relationship between personality and creativity. Creative people are known to have a higher intelligence than average and good predictors of creativity are high verbal fluency, impulsiveness, breadth of interests, independence of judgement and flexibility. They also tend to be able to make associations between different categories of ideas and to be willing to take more risks29.
The study reported here provides an example of a designer who, it could be argued, has a record of creative design based upon the widely acknowledged innovative quality of the artefact itself. In this paper, we focus upon the nature of the human creative process rather than the product or outcome. The study identified some characteristics of Burrows' creative cognition that coincide with findings from other studies of creative people. In particular, Maccoby's investigations into high profile engineers and designers and Roy's studies of innovative product development are relevant\textsuperscript{15,16}.

It is interesting also to compare the Burrows' study with a study of Gordon Murray, the formula one racing car designer reported in Cross and Cross in this issue\textsuperscript{30}. A racing car is a significantly more complex product than a bicycle but, nevertheless, there seem to be similar features in respect of the individual designer's creative scope and cognitive style. The agreement in respect of motivation, personal goals, working from first principles and immersion and expertise across related areas of knowledge is notable. It would seem from this that the cognitive issues in the design process are similar even where the scale of complexity of the artefact, as measured in terms of component number, is far greater.

From such case studies, we learn that creative people have a number of attributes in common. They can be described as "systems thinkers" who enjoy the challenge of identifying new design problems. They are also highly individualistic and there is no single fixed pattern for the way they generate their ideas. In respect of personal motivation, a significant factor is that intrinsic motivation occurs rather than extrinsic motivation: for example, incentives related to remuneration do not figure highly. Thus, self-directed activity is preferred and, not surprisingly perhaps, innovators do not respond to external pressure, in fact this can have a negative effect on commitment. As far as the rewards expected, innovative people enjoy recognition for their work but are not driven by it. They are usually characterised by a sense of play, curiosity and response to the challenge of the unknown. They are, of necessity independent but not solitary whilst preferring not to have to answer to a boss. Nevertheless, they are always seeking new knowledge and aim always to be master of their field. Where they operate with a strong sense of security (nurtured in childhood or by supportive institutions) they function best especially if they are not driven by the requirements of other people.
Mike Burrows conformed to the characteristics of innovative people in many respects. The challenge of unresolved problems is the driving force behind his work and he responds to new opportunities where they arise. He was highly motivated and deeply immersed in his field. When Alex Moulton was asked as to why he wanted to improve the bicycle, he suggested that "perhaps it was the memory of the bike as one's first boyhood means of personal mobility". For Burrows, motivation was expressed as, "I was getting older and I wanted to go faster."

The common factor to note is the source of inspiration in personal experience. Both designers can claim to have revolutionised the bicycle in different ways and, as the history makes clear, without total commitment and motivation, combined with deep knowledge, the results might have been very different.

8.2.4.1 Creative Cognition in Design

Creative people have been described as systems thinkers who enjoy the challenge of identifying new design problems. The challenge ceases when the solutions are identified. In design, they try to see the problem as a whole and then work on the solution details in relation to that whole. In the early stages at least, the design problem is tackled on a broad front and the candidate solutions are defined as a part of a whole system rather than decomposed into detailed parts. This is a form of 'holistic' thinking. The alternatives may be explored by examining a range of options and trying where possible to keep a number of channels open until the best solution presents itself. For example, the car may be seen as a composition of parts for high performance or as a single element in a transportation system. Thus, a characteristic of the innovator is someone who goes beyond the invention itself to consider its impact on a larger system.

In terms of developing ideas, creative people prefer to choose their own problems rather than have them set for them. They look for untrodden fields and pose questions as to why something is not being done in a better way. The best new ideas come from being able to move design ideas from one product area into another. The advantage of being able to envisage solutions to problems drawn from outside the particular domain is that the constraints are not fixed. The benefits of placing a problem within a totally different conceptual space have been argued before. However, this capacity is only beneficial if the
person has been totally immersed in a problem and has the intellectual power to construct things in a new way.

Mike Burrows' approach to design started with familiar (product) models where each step forward was based upon an existing specification. The transformations that took place varied in significance: some small leaps were followed by a major change. However, it is clear that there were important dependencies. In Candy and Edmonds\textsuperscript{18} the development of the bicycle designs is discussed in terms of the product models changes that led to the monocoque bicycle breakthrough.

Moulton argued that "the man who holds the concept and belief in an innovation's feasibility should be made to demonstrate this."\textsuperscript{31} In the case reported here, all stages of the design and building were carried out by the designer himself. From the initial generation of ideas, the rough sketching, tube brazing and component-making in steel and aluminium to the testing in competition, he was the initiator and executor throughout. Later, when the carbon fibre frame making was carried out he engaged the help of others for the mould construction and materials processing. He was the designer, the engineer and the evaluator.

The Burrows' design process was very dependent upon personal ways of working. The style of "designing in the head" combined with "designing in the hands" was essential to his effectiveness. In this scenario, whilst the role of sketching was important\textsuperscript{34}, the development of detailed dimensional drawings was not. The ability to formulate the problem at a high level and devise both systematic and opportunistic strategies for moving forward, were key features because these enabled him to break out of the existing conventions of bicycle design. Designing in parallel areas, such as HPV design, was also very significant in promoting the creative insights that occurred. The innovative design outcome, resulting from years of deep involvement and immersion in the expert knowledge of the field, could not be considered a chance or accidental event.

\textbf{8.2.4.2 Computer Support}

Mike Burrows did not use any form of computer support for his design and engineering. He was able to progress his work within the boundaries of the resources he possessed and achieve a standard that satisfied his personal and business requirements. To go beyond that
to high volume production or to turn-round his designs and customise them faster would have required significant changes in resources and, indeed, in his whole style of working. He was aware, nevertheless, of the potential of computer support and did not have particular inhibitions about using new tools and methods except in so far as they were not suitable for the task. Of a visit to a CAD department he noted,

"They'll show you some wonderful graphics and the stress pattern field. They've got a wonderful big three axis co-ordinate memory machine. There was no design element. They simply just drew it up on the screen and simply changed the shape and then obviously used those to generated the aerodynamic profiles within it rather than just draw a line on a piece of paper, they can obviously program them in. But I wouldn't call that design."

The Lotus Bike case study suggested that existing CAD systems were unlikely to have been helpful in the particular context, not least because of the highly individualised strategies employed by the designer. The need to "design in the hands" was at least as important as designing in the head and there was little use of accurate drawing that is so well supported in current systems. However, the need to work on a team basis led to more extensive use of detailed dimensional drawings for the purposes of conveying accurate ideas and instructions to other members of the team. The key point is that the use of in-depth domain knowledge and access to state-of-the art expertise was critical to the designer's creative thinking. Such attributes would not have been supported or available through existing CAD systems. It could be postulated that providing domain knowledge support to design might be beneficial to designers in general in respect of encouraging more exploratory approaches.

Case studies of creative design extend our understanding of the needs of designer and how they might be supported by identifying the constraints and opportunities that apply. For example, it is important to be able to express a tentative idea and so the support system must be constrained not to insist upon precise information, as is often the case. Equally, the provision of access to diverse and dynamic knowledge sources could provide a significant opportunity for the enhancement of the design process. Some key issues for research into design knowledge support systems are discussed below.
8.2.5 Future Research Directions

In the light of the study reported and its relationship to the general case, we consider the implications for computer support to design. Future research directions are proposed in respect of the key issues concerning interaction with design knowledge. The Lotus Bicycle design, whilst nevertheless having its own unique characteristics, confirms many of the findings from earlier work that have influenced our current research direction. In a previous paper we identified a set of issues that inform the context of support system design. The research directions described below draw upon our earlier studies and other research into designer practice.

8.2.5.1 The Interactive Knowledge Support System

Traditional CAD systems contain geometric modelling information but do not provide the designer with support for the initial formulation of the problem. Where the aim is to provide computer-based support for early design, it implies addressing the more exploratory processes in which the designer is engaged. In addition, the question of how to provide appropriate knowledge during conceptual design arises. Support for interactive knowledge-based design requires fluent interaction between designer and knowledge in a way that does not impede the concept design process. From a computer support perspective, this requires a number of facilities and functions such as, multiple and parallel viewing of information and a flexible interactive dialogue for handling the information. We term such a computer based support environment an interactive Knowledge Support System (KSS). The overall aim is to provide knowledge that will support the designer during the design process itself. As part of our ongoing systems development programme, we have developed a KSS that enables the designer to externalise knowledge without the help of a knowledge engineer and to change the knowledge as the design progresses. Such a prototype system might include design history in the form of predecessor designs, design rules and knowledge editors and access to information and expertise as illustrated in figures 4 and 5 Illustrations of a Design Knowledge Support System.
Figures 4 and 5 Illustrations of a Design Knowledge Support System
8.2.5.2 Interaction with Design Knowledge

Providing support for design knowledge using domain specific computer-based environments raises a number of issues. There are many questions to be resolved about the character of such systems and their relationship to the way designers currently design, in particular the creative and conceptual aspects of that work.

**Design Knowledge Exploration and Evaluation**

The importance of deep domain knowledge which can be explored was identified as a significant issue for the provision of computer-based support to design. Systems that enable designers to draw upon knowledge of product models that may be used for new designs are needed. Knowledge used in the act of designing is dynamic and in order to apply it effectively it must be allowed to evolve during the process. Thus, the designer requires methods for modifying and creating new knowledge as the design develops.

**Design Space and Design Knowledge Access**

The ability to break out of the conventional design space is a characteristic of the creative designer. Providing canned solutions is inadequate. Keeping a number of channels open in parallel, especially where the problem is a complex one, is a necessary part of generating design ideas. Many problems are solved by analogy. Thus, the designer must be able to switch between tasks freely. Interaction with parallel design product knowledge that can be used for evaluation of new designs is required. This places particular demands on the interaction mechanisms. Designers need task specific tools that maximise their scope and control over the system34,35.

**Design Knowledge Sharing**

The designer has individual design knowledge as well as the shared knowledge of design practice. Design strategies are the special tricks of the trade and distinguish one designer's skill from another. The success of innovative designers depends upon the distinguishing contribution that they bring to the field. Our studies of individual and team design activities have led us to conclude that both private and public design spaces must be addressed in computer support environments. It is particularly important to consider the design team that may be involved. Support systems must address the mechanisms for the
communication of that knowledge to other designers. The sharing of knowledge and the implied shared representations is a significant and difficult issue that needs to be better understood.

**Tentativeness and Uncertainty**

Tentativeness and uncertainty characterise many aspects of early design but is not well supported in current CAD systems. One approach to supporting uncertainty is to allow the user to enter ranges rather than absolute values. Another is to attach certainty values. In practice, the expression of tentativeness is often made with a pencil by drawing soft lines. Pressure sensitive pens could offer a similar functionality for computer input. However, the effective handling and sharing of such knowledge, including its match to human expectation, is an outstanding research issue. See, for example, the work of Sun.

**Problem formulation and the emergence of concepts**

This is a fundamental and difficult issue. Methods are necessary to support users in identifying and formulating the design problems to be solved. Very often, once a problem is fully stated, automatic methods can be employed in its solution. A key concept here is the notion of emergence. During the process new concepts emerge. For example, the addition of a new parameter or the conversion of a constant into variable may play an important role in innovation. The research need is to investigate how pattern recognition and representation transformation can be employed so as to anticipate emergence and to facilitate useful interaction between the designer and the system.

**Strategic knowledge and strategy development**

We are currently investigating the fundamentals of the representation aspects of this problem. The designer's strategies are a significant influence on the quality of the results. They distinguish one designer's skill from another. The ease with which the organisation and, in particular, its design section can maintain strategic knowledge, even during the process of investigating a particular design, is an important question.

**8.2.6. Conclusions**

The creative designer is a knowledge worker involved in activities that are not readily characterised by formal procedures. Computer-based design, therefore, must move
beyond well-defined processes, to providing support that is flexible and usable and which complements the thinking processes of the designer. In this paper, we have discussed a study of the design of the LotusSport bicycle, focusing upon the creative cognition of the designer. This work is part of a series of design studies that serve to inform the development of future design support systems for creative work by identifying the constraints and opportunities that operate in a given situation. To that end, we have proposed research directions for future design support environments that include implications for the knowledge intensive aspects of design.

Acknowledgements

The authors would like to thank in particular, Mike Burrows and Lotus Engineering for their support and co-operation. We are also grateful to the reviewers for their constructive comments. The work was partly funded by the ACME Directorate of the SERC, Grant number GR/J41680. In particular, we would like to thank Mike Burrows for permission to reproduce his sketches. The photograph of Chris Boardman riding the Lotus bicycle is reproduced by permission of the second author.

References

1. Blessing, L. T.M. 'A process-based approach to computer-supported engineering design'


10. BBC Television. 'QED The Bike' *Chrysalis Productions Ltd* (1992)


27. Schön, D.A. 'Designing as a reflective conversation with the materials of a design situation' *Research and Engineering Design* Vol 3, No 3 (1991)

28. Gregory, R.L. 'Inventions challenge conventions to create the human mind' *British Association for the Advancement of Science Conference*, Keele (1993)


37. Sun, R. 'Structuring knowledge in vague domains' *IEEE Trans on Knowledge and Data Engineering* Vol 7, no 1 (1995) pp 120-136


9.1 REPRESENTATIONS OF STRATEGIC KNOWLEDGE IN DESIGN


Abstract This paper is concerned with the purpose and form of representations of strategic knowledge in design. A number of issues concerning the role of representations in design are considered. A distinction is made between representations for describing design and representations for guiding or prescribing design. Issues about the role and form of representations "of" and "for" design are considered in the light of experience from a case study of vehicle engineering design. The goals of the project from which this work arises were two-fold: first, to identify examples of strategic knowledge from a domain expert and second, to inform the computer support design field about that issue. The approach adopted was to gather empirical information and to model the results using different analysis frameworks. Three types of representations are described: a domain model of strategic knowledge for supporting interactive system design, a computational model for representing strategic knowledge and a design process model of strategic knowledge in engineering design.

Keywords : strategic knowledge, design, representations, interactive systems design

1. INTRODUCTION

This paper is concerned with the purpose and form of representations of strategic knowledge in design. It draws upon two streams of research that have been undertaken in the MULTIK project (Multi-Layered Knowledge-based Interfaces for Professional Users) [1]. The first stream is concerned with the investigation of the way design takes place, and in particular, the way designers employ strategic knowledge. The second stream is concerned with the development of a logical framework in which strategic design knowledge is implemented using a computational system based upon a multi-layered logic representation. The work combines the contributions of Artificial Intelligence and Human-Computer Interaction to modelling design.

The concepts that are presented in this paper arise from a study of design practice carried out in the context of vehicle engineering design. For any investigation of design in practice, the choice of data collection methods, the analysis frameworks and the representations used will have a significant impact upon the utility of the outcomes. Two specific purposes to which the outcomes were directed were distinguished as follows:-
1. the representation of strategic knowledge derived from qualitative empirical data.
2. the representation of strategic knowledge for implementation in a computational system.
In addition, during the study, a representation of a design process for a specific project that was part of a strategy for reformulating the existing practice, was identified. The purpose of the resulting model was to effect change and to communicate its benefits to the engineering design process.

The goals of the study were two-fold: to identify examples of strategic knowledge in the domain context and to inform computer support design. The approach adopted was to gather empirical information and to model the results using different analysis frameworks. The data collected which consisted of interview transcripts and documents was analysed using two types of representations. The approaches to the analysis and representation of empirical evidence are described.

The study gave rise to a number of issues concerning the appropriateness and effectiveness of the representations used. An existing representation was reviewed and extended and the modifications applied to the resulting models described in this paper.

Three models that have different representational purposes were derived from the study findings. The models, which are described in Section 5 below, are as follows:

i) A model for representing strategic knowledge computationally
ii) A model of strategic knowledge for interactive system design
iii) A model of strategic knowledge for reformulating the engineering design process

The models are discussed in the light of the purpose for which they are intended and the goals of their users.

2. BACKGROUND

In this section, two background themes are discussed. The first is the need to distinguish between representations of design according to the purpose for which they are intended and the type of 'user' who may apply them. The second is the role of representations in modelling the results of empirical studies of design. Both themes contribute to the focus of the work described in the paper, that of deriving appropriate representations for application in the interactive computer systems design and development process.

2.1 Representations of Design: Purpose and Methods for Empirical Studies

The term 'representation' has a range of different meanings. A form of representation may be described as a notation together with an interpretation of the notation [2]. Representations are used for a number of purposes and take many different forms according to who is devising and using them. A great number of external representations
(as distinct from internal mental representations) are used in everyday life e.g. graphs, tables, flow charts and, especially with the widespread use of computer based applications such as spreadsheets and simple databases, have become commonplace personal tools. In professional work, the use of representations such as mathematical formalisms, musical scores, knitting patterns, programming languages, product models, etc., are the result of years of learning and the accumulation of expert knowledge. In these cases, the derivation of a new representation usually indicates a new perspective on existing knowledge or, indeed, new knowledge itself.

In design research, there is a continual quest for representations that provide an understanding of how design takes place and also representations that improve the process of design. Thus, an important distinction is that between representations or models for describing design and representations for guiding or prescribing design.

Whether the representation is of or for design, it should be selected according to criteria such as:

- it should be accurate: include essential features/attributes of the domain or system.
- it should not confuse: aim for clarity but beware of over-simplification.
- the style of representation should be appropriate for its purpose and context of use

Gathering empirical data about what happens in design can be carried out using a number of different methods: e.g. monitoring of design activities by direct observation, structured interviews, unstructured discussions, audio and video recording etc.. The nature of that information, including how comprehensively it covers the activities under investigation, the multiple perspectives of the subjects and the form in which it appears, will affect the kind of analysis that can be carried out and, in turn, influence the way that the results can be used.

Having collected the data, there then comes the issue of the analysis activity. Extracting information that is meaningful depends upon the way it is analysed. For that reason it is important to identify the purpose for which it is intended. If the purpose is, for example, to identify the critical communication patterns that take place in a collaborative design environment, the method used must be able to extract and represent such information. The range of patterns identified will, of course, depend upon how wide and deep the data collection was in respect of formal and informal communication pathways.

Different representations may be derived from the same data set. The choice of representation should be appropriate to the level of abstraction required and the user of the
representation. The analysis methods themselves may influence the form the results take depending upon the situation under investigation. For example, it might be possible to collect what appears at first to be a comprehensive set of quantitative data using a computer-based system about the transfer of designs between different parties to the design in different locations. That data (depending on how the tool is configured) will provide information such as number of items sent, who sent it, from which site and when etc.. The results can be represented as tables or graphs showing the movement of designs over a given period of time. However, if no additional data about the context in which the design transfers take place is collected, this will impose limitations on the interpretation of the data. For example, if there is a gap in time or a change of personnel, the reasons for this will not necessarily be known to the investigator. Thus, unexpected interruptions, such a system failure or organisational changes that restructure the design teams, cannot be taken account of and there is a danger that a false picture will emerge.

For the above reasons, when designing an investigation, the choice of data collection methods, the analysis frameworks and the representations used are critical choices.

Key questions to address are:

- who is the intended user of the representation?
- for what purpose is the representation to be used?
- what form should the representation take?

What is not often made explicit in the representation of design processes as prescriptive or descriptive models is the influence on the model of such factors as:

- the target level of the domain under consideration (e.g. the nature of the granularity, whether designing a building, a drug infuser or a nut and bolt).
- the purpose behind the model (e.g. to teach students how design should take place or to represent the design process activities as the organisational context demands, or to describe how the individual designer works).
- the influence of the domain of application itself on the model (e.g. whether the constituent elements differ significantly from engineering to product design).
- the impact of tools and new methods upon the existing process (e.g. the change elements inherent in the model but not represented explicitly).

It is important to differentiate between representations that are used to support the analysis frameworks and the differing ways of representing the results. Representations may be used for a number of purposes and take different forms according to the domain context, as well as the expertise of those who generate and apply them and according to the intended purpose.
Different representations are needed according to the needs of the intended users of the representation. Representations may be used for different purposes by different people depending upon the roles they have in the process. Some key purposes and representation user groups are identified in Table 1 below:

<table>
<thead>
<tr>
<th>Representation Purpose</th>
<th>Representation User</th>
</tr>
</thead>
<tbody>
<tr>
<td>To describe how design takes place</td>
<td>Researcher/Investigator</td>
</tr>
<tr>
<td>To communicate designs between parties to the design</td>
<td>Designer/Design Team</td>
</tr>
<tr>
<td>To prescribe and control how the design should take place</td>
<td>Design Manager</td>
</tr>
<tr>
<td>To advise/teach how to design the best way</td>
<td>Design Educator</td>
</tr>
<tr>
<td>To represent a model of a user context and requirements for the computer system design</td>
<td>Interactive Systems Designer</td>
</tr>
</tbody>
</table>

*Table 1: Representations: Purpose and User*

For interactive computer system design, two kinds of representations are needed: first, representations of the user needs and tasks to be addressed, i.e. domain models; and second, representations for generating ideas about the design of the system itself, i.e. computational models. A design team manager, on the other hand, requires a representation that will perform a role in guiding/controlling the process and also enable communication within the design team. Examples of all three types are illustrated in section 5 below.

These issues are addressed in the following section with regard to designing interactive computer systems.

### 2.2 Representations for Interactive Computer Systems Design

Interactive computer systems designers are faced with a choice of many representations for design which may be suited to different circumstances. They need to both understand what the user needs and requirements are as well as be able to represent their own understanding at different points in the process. The selection of suitable representations is also necessary for the exploration, generating and evaluating of ideas and decisions within the design team and with users. Thus, the interactive system designer requires both representations of the users, tasks and context and representations for the design of the system.

In both 'design for' and 'design of' representations there will different forms e.g. work flowcharts, user models, functional hierarchies, knowledge-based structures, rules, user interface dialogue designs. A systems design and development project may be progressed using a number of different representations at different stages in the process from the
requirements analysis and definition and task analysis to the system specification and detailed design.

Representations that capture the work practices of a given situation may play a role in informing the computer system designer about the existing task processes but they do not necessarily provide the best means of modelling the computational system to be developed. That will depend on the particular circumstances of the development project. If the aim is to automate parts of the process and replace some human-driven aspects of it, the representation will need to model explicitly the allocation of tasks between humans and machines. To do that effectively, the information gathered must discriminate between tasks best done by human and those best achieved by machine.

A number of limitations to the existing models and representations for system design exist. One problem is that the communication role of the representation cannot be guaranteed. From the originator of the model, it is passed on to others and, especially where there is no direct interpretation, changes of meaning take place. In addition, what is clear to the systems designer is not necessarily readily understood by the user groups and this poses difficulties in communicating crucial design ideas that need to be understood by and acceptable to the user.

Representations of users, tasks and information flow are used to mediate systems design. The representations needed will vary according to the context of use and that part of the system design process to which they are applied. A number of approaches have been developed in Human-Computer Interaction and software engineering for the design and implementation of interactive systems. However, it is clear that there are a number of gaps between the domain modelling that informs system design and the software design expressed as computational formalisms. The matching of a user-centred focus, such as that of interaction design, and a computer system-centred focus, such as functional design or knowledge-base design, has yet to be achieved [3].

The question of how to model the expert knowledge required of a computer support system has traditionally been treated as a area of inquiry that is quite separate from the interactive design issues. Domain models are not readily adaptable to the representation of knowledge in a computational form. There is a need for representations that provide bridges between the user-task-work context descriptions, which are usually expressed in natural language or as flow diagrams, and the formal specifications that are executable. In the following section, the issues are discussed with particular reference to the representation of strategic knowledge.
2.3 Representing Strategic Knowledge

For the representation of strategic knowledge, notations are required that allow the manipulation of actions and dependencies in a non-linear form. Different types of solutions and knowledge require integration of different ways of representation and, therefore, of reasoning.

Another important missing feature in conventional models of design is the representation of changes that take place in the process over time. Most representations are static views of a process and do not easily afford the changes that are almost inevitable. One effect is to restrict the interpretation of what is actually going on: e.g. the enforcement of procedures in a computer system designed on the basis of a hierarchical task analysis model. This limitation is particularly important in the representation of strategic knowledge because, by its very nature this kind of knowledge is expert, dynamic and evolving. A need to give more attention to divergent tasks i.e. tasks that require heuristics for idea generation and evaluation has also been identified.

Further research is needed into the different strategies applied by designers and their impact on the effectiveness of design. A question to be addressed is "why" use a specific approach? Descriptive studies tell us what happens but not the underlying causes: why does design happen the way it does? Research is needed to support the dynamic character of design especially on a strategic level to enable more flexible task-sequencing and knowledge retrieval method. The differentiation between a systematic approach to design required at the organisation or project level and the designer's individual strategies and working practices is not sufficiently drawn out in most design models. The design model should allow different strategies for specific situations.

Expert knowledge and experience play a role in selection of strategy. Different categories of knowledge apply at different points in the process: e.g. declarative, procedural, strategic domain, substantive, general purpose. An important point to note from an examination of the empirical findings about design practice is that the strategies used by the designer are dependent upon the intermediate steps taken in the process of generating a solution and, it follows, therefore, that they cannot be all prescribed beforehand [4].

3. FRAMEWORKS FOR ANALYSING EMPIRICAL DESIGN DATA

In section 3 below, two analysis frameworks for representing the results of empirical data analysis are illustrated. The results from the analysis are shown in section 4.
3.1 Representations of Design: A Domain Model

The first analysis framework is based upon a model of design that is intended to inform computer support tool design. The model itself is based upon both empirical and theoretical work and can be found in Blessing [4]. The domain from which the model is derived is mechanical engineering but other domains and examples of design models and studies were used to inform the thinking behind it.

In Figure 1 below, the Design Process is represented overall as one that is divided into stages, activities and strategies. A brief summary of these elements follows:-

The stages consists of a problem definition stage that leads to a set of requirements. This is followed by the conceptual design stage during which solution principles or design concepts are derived. The detailed design stage then leads to a full product description. The activities consist of items that relate to the individual's problem solving process. These may reoccur and are categorised as types, such as generating, evaluating and selecting, modifying and documenting, collecting information using methods and tools. The strategies are the sequences in which the design stages and activities are planned or executed. Types include: iterative, cyclical, decomposition, stepwise, the choice of which to apply is considered to be dependent on the design context.

![Figure 1 Blessing Model of Total Design Process](image)

The value of this kind of representation is limited in certain key respects. The model is intended to be adapted to allow for differences between problems. Because it is assumed that the overall design process is generic in character, the application domain is not included as an explicit part of the model. Thus, it does not encompass sufficient detail to provide a recognisable picture of the domain context. This has implications for the consideration of the role of a computer support tool. A support system that embodied strategic knowledge might have very different characteristics between domains. Thus, the value of this model with regard to informing tool design remains doubtful. In particular, it is not detailed enough to discriminate between the task allocation of humans and computer
system. It was, however, useful for the data analysis process in giving an appropriate structure to the information acquired. This is illustrated further in section 4.1 below.

### 3.2 Representations for Design: a Knowledge Model

Gruber [5] proposes an approach to representing strategic knowledge in the context of a knowledge acquisition exercise for a medical application. He defines 'strategic knowledge' as knowledge used by an agent to decide what actions to perform next, where actions have consequences external to the agent. Strategic knowledge is a basis for expertise in domains where the management of problem solving depends upon the choice of actions to take and this choice is problematic and, therefore, a matter of considerable human expertise. This kind of knowledge is distinguished from 'substantive' knowledge, which, in a computational system, is used to constrain the search space to the areas that are more likely to contain a solution.

Conventional methods for building knowledge systems depend upon tools that are usable by knowledge engineers not domain experts. Such tools may impose a general purpose procedure for applying rules and the knowledge engineer's task is to select the rules for firing using primitives such as numeric priorities. Gruber's goal is to acquire strategic knowledge from experts without assuming that they use the same symbolic representations within a computational system. To that end, there are two key issues: the elicitation of tacit knowledge and the representation mismatch between the form of knowledge found in the expert person and its form in a computational system.

The conventional approach to knowledge systems development is not useful for acquiring domain expertise especially strategic knowledge. Interfaces are needed in order to enable the direct evaluation, modification and augmentation of the knowledge base by the domain expert. This is a 'reactive' approach to strategic knowledge in the sense that no look ahead or planning is assumed and the expert judges which steps to take on the basis of the immediate predecessors. This has been studied in previous work by the author [6,7].

In the current work, an approach to acquiring and representing strategic knowledge has been adopted following Gruber [5]. The process involves three main stages as follows:

i. Acquire strategic knowledge direct from domain expert

ii. Design the representation of strategy rules

iii. Design an interface for the human-knowledge system dialogue

The representation of the rules for strategic knowledge is described in 4.1 below. This paper addresses the first two steps and the third is being addressed in related work [1].
In the following section the results from applying the analysis frameworks to the case study data, based upon the models described above, are presented.

4. **A STUDY IN VEHICLE ENGINEERING DESIGN**

A study was carried out which investigated the principles, practices and strategies applied in vehicle engineering design, including the human and organisational issues. The objective was to identify strategies used at different levels of the total process from customer requirements to product specification. The outcome is a set of findings that are relevant to design modelling and interactive computer systems design.

The approach adopted was to gather qualitative data from an informant who was a highly expert senior design engineer. Existing design models were then considered in the light of that information. The design process models were included as reference points in the data gathering about how design was carried out in the domain context of vehicle engineering design. A structured interview questionnaire was devised, based upon A.D.E.CT (Automotive Designers' Ergonomic Clarification Tool set) [8]. It was modified to take account of the specific goals of the particular domain and organisational context. These models were drawn from the Design modelling literature across a range of disciplines.

The subjects covered in the interview questions were as follows:-

- Models of the design process
- Examples of design projects
- Market/Customer requirements
- Product Design Specifications
- Concept Design/ Detail Design
- Computer Support Systems

The interviews were recorded by audio cassette tape and transcribed. The resulting qualitative data consisted of several interview transcripts and other material such as published papers and company documents which were drawn upon during the interviews and follow up feedback discussions.

The data was analysed using the analysis frameworks described in section 3 above. The transcriptions were analysed in two stages:-
1. Three independent annotations of the transcript were carried out.
2. Drawing upon the results of 1. a second analysis was carried out using a different set of categories that were intended to clarify the issues further for feedback to the informant.

4.1 Analysis Frameworks Applied

Two types of analysis were performed on the data gathered. The data was analysed using a representation of design (after Blessing [4]) and a representation for design (after Gruber [5]) as described below.

The first stage in the analysis was performed by applying the Blessing Model (see Figure 1 above) to the transcript analysis. This was carried out by asking by three evaluators to independently assess the transcript information in accordance with the categories as defined. This involved deciding first, those aspects which could be defined as design process stages, activities or strategies and second, those which were individual, team or organisational matters.

This analysis gave rise to findings about the organisational design process and team and individual design strategies. The representation of these results were provided to the domain expert in order to check whether they were accurate and clear.

For the second analysis framework, data was classified in such a way as to provide an initial representation that could in turn be expressed in a computational form. The second type of analysis was carried out according to three main categories based upon a rule structure designed as follows. The categories were as follows:-

- **Classification**: design process characteristics
- **Pre-Conditions**: prior existing factors
- **Strategies**: actions taken in relation to pre-conditions

*Classification* refers to a characteristic of the design process: for example, requirements, concept design, detail design, business success, design quality.

*Pre-Conditions* are factors that have been identified as existing prior to certain actions taken. The pre-conditions have implied "ands" but they could be "ors". It might be that a single pre-condition automatically implies the others listed.

*Strategies* are actions taken in relation to the pre-conditions. As in the case of pre-conditions, there are implied "ands" and "ors".

304
The data was analysed using this rule structure. The results were a set of propositions that were firstly used to check the findings with the informant in order to refine that knowledge further. See Figure 2 in section 4.2.2 below, for examples.

This approach is aimed at developing a computational representation of knowledge, in particular meta-level knowledge.

Both of the above kinds of representations were applied to the analysis of qualitative empirical data in the study described below. The analysis resulted in a set of propositions that were used to check the findings and to gather further knowledge from the informant.

4.2 Results

The results of the study gave rise to a number of conclusions about how the design process was carried out in the given domain context. The findings ranged from information about how the domain expert managed the design process and his design teams to specific project processes applied to individual components. In the following sections, example results from the two types of analysis are provided. The results from this exercise were then applied in the derivation of models of and for specific design processes as described in section 5.

4.2.1 Examples from Analysis Framework 1

Examples of how high level strategic knowledge were considered within the overall design process were identified, some of which are provided below.

- **Innovative versus Evolutionary Design Strategies**
  Factors that affect the choice of design representation are whether the design scope is truly innovative, is without significant down-line constraints at the outset, or whether it is of an evolutionary kind which requires the importation of many existing systems that are to be customised and improved incrementally for the particular project.

- **When to apply constraints**
  If it is possible to start with a large set of candidate solutions, an idea might emerge that is promising enough to pursue. However, the practical problems of large scale complex product design do not often afford opportunities for this approach because of cost factors. By omitting down-line considerations in certain cases, this assisted the generation of new solutions to a new problems.
• **Solution-Led Strategies Vs Problem-Led Strategies**

A design solution is difficult to derive from a requirements list without other points of reference: e.g. candidate solutions or knowledge of domains where solutions might lie. This Problem-Led Strategy requires a stage for generating candidate solutions and/or bringing in domain knowledge at an appropriate point. Too many solutions is hard to handle in an intellectual or cognitive sense for one person and the contribution of a larger team is essential to the process of generating the list of all conditions against which to test the proposed solutions. A Solution-Led Strategy may apply more to individual designing whereas a Problem-Led Strategy works better in team design.

• **Use of Knowledge and Experience in Solution Selection**

In order to generalise at a high level, considerable experience of the field as well as the ability to pursue further research is needed. Having a personal strategy for design decision making is born out of experience. For example, knowing how to reduce the list of candidate solutions quickly to a manageable size requires prior experience and knowledge of those items that have minimal and maximum impact.

• **Expert Knowledge and Prototype Evaluation**

Prototypes are useful mainly to evaluate particular attributes or components of the design. An expert filters the quality under consideration and evaluates it against a backdrop of inferior features. This kind of expertise is a valuable asset to a company and enables it to maximise the value to be gained from prototype building.

• **Team Design**

An example of a team design strategy for bringing engineering knowledge to bear on design styling decisions is to allocate engineers to the vehicle styling team. Where styling poses heavy constraints on the mechanical engineering, it may give rise to expensive or lower quality solutions. By placing feasibility engineers in the styling stage, this is a strategy for bringing engineering considerations to bear upon early decisions at a point when alternatives exist and when it is possible to agree a best compromise.

• **Individual Designer Strategy**

An example of an experienced individual designer's strategy for addressing a complex problem area is to discriminate between high and low impact factors early on in the design process. Designers who are familiar with the problem area, identify what they consider to be the important aspects of the new design and changes from previous designs, and then "build confidence" in those areas. This establishes the overall picture. Decisions about less critical areas are suspended for the time being on the assumption that they can solve them.
when the time comes. For the experienced designer, this is not a high risk strategy for finding a solution, but it increases the possibility that the design solution will be less good, i.e. inelegant or expensive.

The above language descriptions of strategic knowledge were identified by applying the elements of Blessing model to the data. However, whilst informative and useful to an understanding of domain expertise, they do not provide a structure which offers a natural means to check and extend the information, nor is it conveniently expressed in a computational form. The next step was to identify classifications and derive appropriate representations of the knowledge.

4.2.2 Examples from Analysis Framework 2

In Figure 2 below, the empirical data has been categorised according to its design process classification, the pre-conditions that apply in the given context and the strategies taken to progress the design project. The example illustrates the key strategies and their pre-conditions in relation to different classifications of the design processes.

The context is one where an engineering service company has been commissioned to improve the suspension design of a luxury vehicle by a large vehicle manufacturer. In the example shown, certain pre-conditions exist. Those pre-conditions range from having available the human expert knowledge on site and recent prior experience that combine to underpin the actions to be taken, to the target niche market where amongst the known competitor vehicles, the differences are nuances. In that context, the client's requirements are also pre-conditions, for whilst the existing product's best qualities must be maintained (‘protect good features’), the suspension system which must be improved (the problem is ‘x’ : e.g. ‘shake’) has qualities which are fundamental to the individual character (‘distinctiveness’) of the vehicle.

The strategies listed in the right hand column, begin with the overriding goal of testing the existing design against a set of criteria and then taking steps to improve it. In order to determine what those criteria should be, the approach is to adopt the prospective vehicle owner’s point of view and to carry out subjective evaluation exercises of all the existing competitors with a range of expert and non-expert drivers on a variety of road surfaces. Once the qualities have been distilled, the next action is to identify the underlying system principles that give rise to those qualities. This is decomposed into the suspension system characteristics and the relationships between components and because a number of different qualities are under consideration, the inevitable trade-offs between adopting certain combinations arise. To handle conflicts, the strategy is to configure the design in such a way as to minimise the interactions between system characteristics.

307
The strategies can be further decomposed into more and more detail and as such can be represented computationally as illustrated in section 5 below.

<table>
<thead>
<tr>
<th>Classification</th>
<th>Pre-Conditions</th>
<th>Strategies</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design Process:</td>
<td>Research knowledge</td>
<td>Test design against criteria</td>
</tr>
<tr>
<td>A Project Process</td>
<td>Prior experience</td>
<td>then change...improve</td>
</tr>
<tr>
<td>Niche market: luxury</td>
<td>Complex problem</td>
<td>Adopt owner's view: language</td>
</tr>
<tr>
<td>Requirements</td>
<td>Distill differences from competitor products to provide right blend features</td>
<td></td>
</tr>
<tr>
<td>Nuances separate products</td>
<td>Identify basic principles of systems/products</td>
<td></td>
</tr>
<tr>
<td>Maintain existing product distinctiveness</td>
<td>Identify product qualities</td>
<td></td>
</tr>
<tr>
<td>Requirements</td>
<td>'X' is a problem (e.g. shake)</td>
<td>Think high level</td>
</tr>
<tr>
<td>Protect good product features</td>
<td>Identify basic system principles</td>
<td></td>
</tr>
<tr>
<td>Conflict resolution</td>
<td>Distill differences from competitor products</td>
<td></td>
</tr>
<tr>
<td>Subjective evaluation results conflict and people are well calibrated</td>
<td>Identify relationships of characteristics</td>
<td></td>
</tr>
<tr>
<td>Conflict resolution</td>
<td>Conflict resolution</td>
<td>Test ideas: apply analysis</td>
</tr>
<tr>
<td>Candidate solutions have conflicting strengths &amp; weaknesses</td>
<td>Protect high level decisions</td>
<td></td>
</tr>
<tr>
<td>Design Quality</td>
<td>Follow two routes in parallel</td>
<td></td>
</tr>
<tr>
<td>Customer requirements</td>
<td>Protect high level decisions</td>
<td></td>
</tr>
<tr>
<td>Product qualities are x,y,z....(subjective)</td>
<td>Conduction evaluation team exercises</td>
<td></td>
</tr>
<tr>
<td></td>
<td>- use subjective feedback</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Optional extra: client clinic tests</td>
<td></td>
</tr>
</tbody>
</table>

*Figure 2: Example Results applying Analysis Framework 2*
5. THREE MODELS OF STRATEGIC KNOWLEDGE

The vehicle engineering design study raised issues about the appropriateness and effectiveness of existing design representations. A majority of the design process models that were used as comparisons or reference points, were found to be contingent with the particular domain and recognisable in practice by the expert informant. However, no single model was appropriate to a majority of the design projects or processes in the expert’s considerable experience. After due reflection on the results of the study and the appropriateness of existing representations, three new models were derived.

The three models that represent strategic knowledge in design are presented in sections 5.1 to 5.3. They are as follows:

A model for representing strategic knowledge computationally
A model of strategic knowledge for interactive system design
A model of strategic knowledge for reformulating the engineering design process

The first example, a computational model of strategic knowledge, applies the NASK concept [10]. NASK is an extension to the Prolog interpreter which incorporates ‘negative knowledge’ and an automatic asking mechanism (i.e. for asking the user when the system does not know whether it is true or false). Negative knowledge is expressed as explicit statements where something is known not to be true. The model is described in 5.1 below.

The second and third models were represented using the MULTIK model [1] and are described in 5.2 and 5.3 below. The first MULTIK model is described in Clibbon et al. [9]. The model was extended to include 'Actors' which represent the role of designers and design teams and computer tools in the design process and Connectors which represent activities carried out by specific Actors. The complete model consists of three main elements:

- **Objects**: representing those items that are inputs or outputs to the design process: e.g. requirements lists, product models, organisational principles and working practices, targets or criteria, design layouts and mechanical arrangements

- **Connectors**: representing those activities that occur between object inputs and outputs to the process, that might be conducted by automatic means (i.e. computer or machine), by human beings as individuals or as a team or by the organisation or company. The connectors are labelled to indicate the nature of the activity. They do not imply any particular sequence or dependencies.
• **Actors**: representing the *human and tool roles* and their different combinations that drive the activities.

The purpose of the model is to represent the different entities that operate in a complex systems engineering environment in a way that distinguishes clearly between the key elements of the total process. It was used to build the models of strategic knowledge in design with a view to informing the design of computational support systems.

The models are presented as examples of how certain representations are more suitable than others depending upon the intended purpose and user and the domain context.

### 5.1 A Model for Representing Strategic Knowledge Computationally

This example is drawn from the data analysed using Analysis Framework 2. Having acquired examples of strategic knowledge from the vehicle design domain expert, the next step was to represent the strategy rules. A subset of the results were expressed as a set of propositions about the design process according to the second analysis method. These propositions were confirmed as an accurate set with the expert.

The data was structured according to Classifications, Pre-conditions and Strategies. Strategies are actions taken in relation to the pre-conditions. An example of the representation of strategic knowledge using this method and derived from the empirical data is shown in Figure 4 which extends that of Figure 2 in section 4.2.2 above.

The next step was to take this and construct computational representations so that it could be applied in a design support system.

<table>
<thead>
<tr>
<th><strong>Classification</strong></th>
<th><strong>Pre-Conditions</strong></th>
<th><strong>Strategies</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>Design Quality</td>
<td>Customer requirements&lt;br&gt;Product Qualities are x,y,z...(subjective) &lt;br&gt; If subjective quality is: &quot;Good Ride is defined as absence of bad ride quality&quot; and &quot;Ride quality is acceptable to (P) people &lt;br&gt; Road conditions are a,b,c..</td>
<td>Follow two routes in parallel&lt;br&gt; 1. Identify quantifiable aspects of qualities&lt;br&gt; 2. Conduct evaluation team exercises&lt;br&gt; - use subjective feedback&lt;br&gt; - run client clinics-optimal extra &lt;br&gt; Adopt Y solution bumpy, smooth, off road. Adopt Z solution for all road conditions</td>
</tr>
</tbody>
</table>

*Figure 4: Classification, Pre-Conditions and Strategies in Vehicle Design Knowledge*
A computational representation of strategic knowledge may be applied using the concept of "asking the user" together with the concept of negative knowledge [10].

An experiment was carried out using NASK as follows. The code represents the knowledge as classified in Figure 4 above. It should be noted that, for illustration purposes, it is an interpretation of only part of the strategic knowledge implied in the table.

The following is the source code for representing the knowledge.

```
design_car:-requirements(X), design(X).
design(X):-subjective(X), (objectify(X); team_eval(X)).
design(X):-objectify(X), confirm(X).
objectify(X):-quantify(X,Xq), confirm(Xq).
team_eval(X):-set_people(P,X), acceptable_to(P,X).
```

The following is a sample dialogue showing the interaction of a user with the knowledge on execution of the above by NASK.

```
User : design_car
NASK : ?requirements(X)
User : X is good_ride
NASK : ?subjective(good_ride)
User: true
NASK: ?quantify(good_ride,Y)
User: do_not_know
NASK: ?objectify(good_ride)
User: do_not_know
NASK: ?set_people(Z,good_ride)
User: Z=25
NASK: ?acceptable_to(25,good_ride)  * Perform experiments
User: true
NASK: design_done
```

Footnote * At this point in the dialogue, it is necessary for the user to pursue other tasks to ascertain whether the ride quality is acceptable to the given number of people. This might involve moving into a different module in the design support environment, or using a different application and performing experiments with drivers, before returning to the strategic knowledge module to continue the dialogue.
The exchanges as shown do not represent the actual dialogue that the user would experience. The users interact with a windows-based interface and the level of representation shown above is, therefore, hidden from them. As it is represented in Figure 2 and 4, the knowledge still contains ambiguities. By expressing the knowledge computationally, this proved to be an effective means of identifying which ambiguities existed. However, in real practice, the process of disambiguating this knowledge can only be achieved by providing the expert with an interface to the knowledge and a means of testing it using context knowledge. In an extension of NASK, the system was applied to distributed knowledge bases. It is envisaged that this approach is likely to be a more typical application of the method [11].

5.2 A Model of Strategic Knowledge for Interactive System Design

The second example is of the project design process of a steering column connector. In this example, a large set of candidate solutions were considered and presented to the client. These were pruned by imposing restrictions arising from client feedback.

"they set out the requirements and we went back with a whole load of ideas and then they said we don't like some of these ideas for these reasons and we said but you didn't tell us that before.......they had not thought whether springs were valid or not until we proposed springs. Whether they could have done or not is a different question but they didn't put it in the original requirements."

In respect of the springs solutions, "a spring is not fail safe because it could break", this was an example of a "risk judgement" where a guiding principle overrides a pragmatic engineering judgment.

"one designer might say well if I use a spring with a certain percentage of its proven capability then its all right but others might say we don't accept that the principle of a spring is valid."

The steering column connector project process is outlined as follows:-

- Requirements
  A steering column that connects to the steering mechanism in a single action
  Safety critical item: condition to be maintained to as safe as existing solution
  Must be easily checked
  Must be cost effective with minimum component count.

- Concept Design
  Generate many ideas (things that snap together e.g. springs or wedges)
  Consider all possible mechanical arrangements (include service/replacement conditions)
Analyse and evaluate mechanical arrangements

- Evaluation of Concept Design
  Refer candidate solutions to client for assessment against requirements
  Iterations here.
  Client specification becomes tougher as more requirements are introduced.
  Agree one solution
- Detail design
- Build
- Make one prototype
- Deliver and demonstrate to client

The project process can be represented using the MULTIK model as shown in Figure 3 below.

**Domain Model: Steering Column**

In Figure 3, R1 and R2 are client requirements and S1 is the proposed solution. The intermediary solutions are shown as multiple objects. As the transcript extract shows, when the first set of solutions was presented to the client company by the engineering services company, it was only at this point that certain candidate solutions were ruled out and thus, the design space was further constrained.
In the example shown in Figure 3, the design space is that of the Requirements Layer where concept design is represented. The concept design solutions are pruned until only one or two candidates are left. Once a single proposed solution is accepted by the client, it is transferred to the Object Layer of the model (see Figure 5), where the design is successively refined into detailed component design. The possibility of movement back and forth between Requirements and Object Layers of the design space always remains open.

5.3 A Model of Strategic Knowledge in Engineering Design

The third example is a model of the suspension system design process for reformulating a project process. It is a model of strategic knowledge in engineering design.

During the study, a new project process for suspension design was identified. This was tested in the company and found to be successful. A hypothesis was put forward that the general principles and practices might equally be applicable to other elements, systems and components of vehicle design within the company.

The aim of the exercise was to introduce a project process within the company that would have the effect of achieving more cost effective and quality result that met customer wants. The strategy was to devise a new process, try it out in practice and evaluate the result. On the basis of the experience, this was then disseminated to the rest of the company's design community.

This model was developed during the study as part of the domain expert's search for a method for reforming the existing process. In effect, it is an instantiation of an earlier model of the role of expertise and knowledge in the design process. The earlier model is discussed in detail in Hogg et al. [12].

The engineering system design model (see Figure 5 below) was devised on the basis of experience and knowledge and tested in practice before dissemination to other projects. It was intended to be a model for guiding the team in developing a new project process. The aim was to establish targets, expressed as criteria derived from a combination of objective and subjective tests, against which the design could be tested.

In this model, the subjective assessment, which includes evaluation of parameters such as ride handling, for example, takes place at the outset of the whole process as part of the benchmarking task. There is no reference to 'concept design' as such; instead the
equivalent here consists of a series of stages from setting the targets, establishing system qualities and then system characteristics before any geometric layout design is carried out.

The model appears not to assume fine tuning according to experience at the concept design stage but incorporates subjective assessment based on factors such as response to real road conditions from the outset and does not leave it to the fine tuning stages. The innovation is to bring these evaluation skills to the front end of the design process in the subjective assessment which feeds into the benchmarking. This is used to contribute to the benchmarking against which the system qualities and characteristics will be evaluated. The effect is to delay the point at which layout design begins. If the objective targets have already been met, the layout design should be straightforward and directed towards a single solution.

In both this and the earlier company model [12], the role of the designer's experience and knowledge is crucial. In the previous model, these factors are represented as continual inputs to the whole process. In the later version, the designer's experience is represented as coming into play at particular critical points where the need for interpretation of complex factors is most acute.

The project process is illustrated in Figure 5 below. The original model appears in [13].

---

**Figure 5: A Model of Engineering System Design**
6. DISCUSSION

In the first example, the goal was to provide a computational method for representing strategic knowledge in a rule-based system. The initial representation in the form of classifications, pre-conditions and strategies proved helpful in confirming the findings with the domain expert and was a step towards the computational representation. This representation provides rules for moving around the domain design process. Whilst the rules were not initially expressed in a form that was executable, a further step turned them into computationally readable entities and an interactive computational mechanism was demonstrated.

The second example model, was devised in order to describe the whole process in terms of the procedures, actors and activities and the connections between them. It is being applied further to provide the interactive systems designer with a way of modelling the task allocation between humans and computer tools. It also shows how the process may move back and forth between the different tasks in the Requirements and Object Layers of the design space.

The third example model is applied to reformulating the process model for a specific vehicle engineering product. The customer wants are expressed as objective targets, a feature that is not shown on most models. By introducing benchmarks for evaluation early on, the process is made more cost effective. The domain expert used the success of this model in practice to show how it could be applied to other projects to good effect. An innovative aspect is the delay imposed on the onset of geometric layout by the setting of objective targets and criteria for testing system characteristics that have been derived from the previous subjective and objective benchmarking exercises. It was also useful for explaining to customers how the company ensured quality and cost effective design. Both this model and the second example have been expressed using the MULTIK model which represents the key elements of the design process in terms of human and computer actors, the input and output objects of the process and the connecting activities.

The MULTIK model is an effective way of representing the project process because it can differentiate between the activities and also can associate the activities with specific actors. The overall progression of the design process takes place in a design space where the objects, activities and connectors are shown in relation to one another but not necessarily sequentially.
The reason for representing explicitly the elements of the total design process in terms of objects, connectors and actors, is that it is essential that they be included in any model that is intended to support the design of a computer support system. It is then possible to identify those elements that are appropriate for inclusion within the support environment as a file or module or component, as distinct from those that are outside the computational system and within the scope of human or organisational activity. The actors are attached to activities only but not the objects which are the inputs and outputs throughout the total process. The role of human expertise, both as individuals and in teams, is shown as well as human expertise in conjunction with computer tools. An important distinction in respect of the kind of computer use is that between automatic computer and human guided use. The role of human interpretative skills and experiential knowledge separately from human adjunct to computer use is included. This key element of the total design process is not always given sufficient representation in current process modelling work.

7. CONCLUSION

This paper has described an exercise in analysing qualitative data and representing the results of that analysis. The experience gave rise to reservations about the utility and usability of existing representations of design processes that are in use and alternative models were derived. Strategic knowledge has been considered from three points of view:

- A computational representation of strategic knowledge that has been shown to assist with the implementation of interactive support systems
- A model of strategic knowledge that assists with the design of interactive support systems
- A model that supports the understanding, communication and development of the engineering design process

The models are different in purpose and form; however, they are all applicable in the interactive systems design process, albeit at different stages and levels. The purposes are domain task modelling, computational modelling and process improvement, all of which are relevant to the total design process and will be usable by different members of a large team.

ACKNOWLEDGEMENTS

The author would like to thank K.J. Sears, Lotus Engineering his contributions to the vehicle engineering knowledge described in this paper. Support for the NASK examples and valuable comments on an earlier version of the paper were provided by Ernest Edmonds. The research was partly funded by the EPSRC Grant ref no.GR/J43769.
REFERENCES


318
APPENDIX I FULL PUBLICATIONS LIST

Italicised papers are included in the thesis


ISSN: 1257-8703.


L. Candy (1983). A project which investigates using a computer in English language teaching. CAL (Computer Aided Learning) NEWS 23, August .


Selected LUTCHI Reports


323
APPENDIX 2 A NOTE ON AUTHOR CONTRIBUTIONS

Of the fourteen papers presented for the thesis, all bar one are first author papers and five of those are authored by the candidate only. Nine were published in academic journals and six are refereed conference papers, one of which (2.3) was later published in a journal and another (5.2) as a book chapter in extended form.

For nine papers, Edmonds is a named author. He was the research director of a number of the funded projects in which the author’s case studies were carried out. Edmonds’ contribution to the research agenda and the guidance received throughout is acknowledged.

The details of the paper authorship follow:

**Single author papers : 2.1, 3.2, 4.1, 4.2 and 9.1**


**Papers with Ernest Edmonds as named author : 2.2, 3.1 6.1, 8.1, 8.2**


Other papers: 2.3, 5.1, 5.2, 7.1


For this paper, David Patrick, an industrial placement student who worked under the author’s direction, is a named author. He implemented the VPKSS software described.


The named authors were members of the large team concerned with all aspects of the project. Edmonds is first author of this joint industrial/academic paper which arose primarily from the user study carried out by the author.


The named authors are drawn from the LUTCHI team of the FOCUS project. The author, who drew together the two main streams of work in this paper, was responsible for the evaluation strategy to which Nick Rousseau’s work on the methods was a central contribution. Ernest Edmonds, Susan Heggie and Bryan Murray were responsible for the system architecture and implementation of the software tools. The last gave invaluable feedback on an earlier draft of the paper.


For this paper, Stella O’Brien, was the subject of the participative research study and the speech domain expert member of the project team.
APPENDIX 3 CRITERIA AND SOURCES

In this appendix, criteria with referenced studies are listed. It provides the source evidence that was used to inform the criteria modelling approach described in Chapter 3. The domains in which the studies were conducted included: vehicle styling, engineering design, architecture, product design, mechanical engineering, urban design, speech science etc. The source ascribed to a particular criteria is not necessarily the only one to which it applies.

The criteria do not represent all of the findings from each of the studies cited, nor have they been exhaustively compared. It is a much larger task than has been attempted here to take the evidence and derive categories of criteria for all cases. The findings are drawn from a variety of situations that have been investigated using different methods. In addition, studies which use the same raw data for analysis are very rare, an exception being the collection of studies in Cross et al, 1996.

GENERAL CRITERIA: CREATIVE PROCESS ACTIVITIES

There is continuous and often rapid iteration between creative process activities and thus a clear discrimination between the elements is not apparent. The general criteria presented may apply most often to a particular activity but may also apply to other stages in the process. Hence, there is a need to take an holistic view of the creative knowledge process.

Support Systems for Creative Knowledge Work in Design should allow the user to:-

<table>
<thead>
<tr>
<th>Source Reference</th>
<th>Criteria</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lawson, 1993</td>
<td>Keep several avenues of exploration open in parallel</td>
</tr>
<tr>
<td>Mednick, 1963</td>
<td>Make ‘remote’ associations between different areas</td>
</tr>
<tr>
<td>Visser, 1993</td>
<td>Make analogies with parts and products in other areas</td>
</tr>
<tr>
<td>Candy, 1996</td>
<td>Examine options in different frames of reference</td>
</tr>
<tr>
<td>Candy et al, 1993</td>
<td>Compare knowledge sources</td>
</tr>
<tr>
<td>Candy &amp; Edmonds, 1994/96</td>
<td>Transfer ideas from one design or product area to another</td>
</tr>
<tr>
<td>Candy, 1997</td>
<td>Use personal strategies/Apply team strategies</td>
</tr>
<tr>
<td>Cross &amp; Cross 1996</td>
<td>Design from first principles</td>
</tr>
<tr>
<td>Sharpe- 1994</td>
<td>Consider several options at once/avoid premature closure</td>
</tr>
<tr>
<td>Candy et al, 1995</td>
<td>Move to a different set of contraints: material, cost base,</td>
</tr>
<tr>
<td>Fischer, 1993</td>
<td>Exploit lessons from failures</td>
</tr>
<tr>
<td>Candy, 1996</td>
<td>Have ready access to communication with other experts</td>
</tr>
<tr>
<td>Visser, 1994</td>
<td>Retrieve existing plans and deviate from retrieved plans during designing</td>
</tr>
<tr>
<td>Visser &amp; Hoc 1990</td>
<td>Conduct tasks in opportunistic order vs hierarchical sequence</td>
</tr>
<tr>
<td>Logan&amp;Smithers, 1993</td>
<td>Employ new ideas on exploratory or experimental basis without irreversible commitment</td>
</tr>
<tr>
<td>Wognum, 1996</td>
<td>Retrieve and adapt existing design cases</td>
</tr>
<tr>
<td></td>
<td>Allow solution design under development to be suspended at one level and pursued at another</td>
</tr>
<tr>
<td></td>
<td>Support resumption of a suspended task</td>
</tr>
<tr>
<td></td>
<td>Keep a trace/record of previously abandoned task elements</td>
</tr>
<tr>
<td></td>
<td>Do not impose the timing of a resumption of a task</td>
</tr>
</tbody>
</table>
Support Systems for Creative Knowledge Work in Design should allow the user to:

Source Reference Criteria

Fricke, 1992 Carry out intensive analysis by asking questions,
Organize an information search
Set up variants as distinct from generating too many solutions
Retain an overview in order to handle reduction of solutions
Place initial emphasis on difficult problem areas

Candy, 1997 First tackle problems that affect overall constraint

Ullman et al 1988 Employ long range procedural planning and correct weighting of problems

Schön, 1991 Explore problem and solution together

Marples, 1961 Use alternative solution conjectures as a means of developing the problem

Cross, 1992 Understand the problem

Sharpe, 1994 Provide support for generating a range of alternative solutions

Levin, 1965 Impose additional constraints that narrow the solution space

Akin, 1979 Change goals and adjust constraints during process of designing

Waldron, 1988 (during solution generation)

Tovey, 1992 Use envisage solutions/concepts as a whole
Synthesize ideas in a solution-focused manner
Consider solution proposal as a whole first, work on details later
Use "influence boards" to kick imagination into life as a stimulus to generation of new ideas

Murry et al, 1993 Have facilities for creating graphical objects or icons in addition to pre-specified objects that can be used to express knowledge about domain specific features in order to accommodate the emergence of new concepts during the process.

McNeill, 1994 Switch between activities fluently and quickly.

Calvert et al, 1993 Use visual interaction techniques based upon domain models
Interact with sources using domain specific notations
Draw on domain-oriented notations appropriate to the task in hand

Nardi, 1993 Return to status and reformulate the design in a flexible manner

Gaines, 1990 View, browse and manipulate several knowledge sources at once
Have access to finely graded views of high quality data
Have multiple views of same data drawn from knowledge sources
Have alternative representations of the same data or information
Have facilities for annotating data drawn from knowledge sources
Be able to switch between knowledge activities fluently and quickly.
INTERACTION WITH KNOWLEDGE

The following criteria are drawn from number of studies that investigated the role of knowledge in design: e.g. Lawson, 1993, 1994, Candy et al, 1993, Visser, 1995

Support Systems for Creative Knowledge Work in Design should allow the user to:

Apply different types of knowledge: domain, context, strategic.
Draw upon heterogenous sources: case histories, existing designs, physical models, conversations with other designers, technical journals etc
Have access to knowledge as a set of constraints in the form of domain specific rules that can be used to test and generate design ideas as they are being developed.

Have access to facilities for the consultation of rules in the knowledge base that allows the designer to consider all the elements of the design under consideration and whether these meet existing rules. Where they do not, the designer may alter the rules by adding new constraints or modifying existing rules. In this way the knowledge may be refined and extended. If a support system employs design rules, direct interaction with that knowledge in the system is needed for modification and extension.

Source Reference Criteria

Candy et al, 1995
Analyze the design in progress against a given set of constraints.
Observe or ignore constraints as the design proceeds.
Make explicit changes to the knowledge in the system
i.e. to alter the basic status of the design constraints.

Lawson, 1993
Have support for the analysis and formalisation of knowledge

Candy et al 1993
Gain understanding and expression of knowledge structure
Enable re-assessment of knowledge
Develop new strategies
Have a plurality of representations so that new knowledge that emerges as a result of changes in the user's understanding can be readily incorporated into his or her activities.
Be able to create and evaluate selected rule sets with different variables against identical experimental criteria allowing immediate comparisons of their performance relative to each other.
Have concurrent access to the different forms of visual data and the methods for knowledge base interaction
Evaluate knowledge using rapid feedback
Have facilities to express knowledge directly to the system
Control the rule base structure
Have support for evaluation of the rule base constituents
Hold a number of issues in parallel re content, structure of rules.
Handle increasing complexity as the knowledge is extended.
Have test facilities for validating the rules/knowledge.
Create new rules for exploratory thinking in order to gain more information
Have support for decision making using both negative and
Support Systems for Creative Knowledge Work in Design should allow the user to:

**Source Reference Criteria**

<table>
<thead>
<tr>
<th>Author(s)</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lawson, 1993</td>
<td>Have access to facilities for capturing knowledge dynamically as the design proceeds and restructure to personal style. Have a plurality of representations so that new knowledge structures that emerge as a result of changes in the user's understanding can be readily incorporated into his or her activities.</td>
</tr>
<tr>
<td>Candy et al, 1993</td>
<td>Have a plurality of representations so that new knowledge structures that emerge as a result of changes in the user's understanding can be readily incorporated into his or her activities.</td>
</tr>
<tr>
<td>Calvert et al, 1993</td>
<td>Have multiple sources of knowledge about a range of issues that may or may not be useful to the evolution of new design.</td>
</tr>
<tr>
<td>Bruderlin, 1996</td>
<td>Interact with several layers of information.</td>
</tr>
<tr>
<td>Candy &amp; Edmonds, 1993, 1994</td>
<td>Draw upon heterogenous knowledgesources: e.g. case histories, existing designs, physical models, conversations with other designers, technical journals.</td>
</tr>
<tr>
<td>Gregory, 1992</td>
<td>Have access to a number of prototypical designs as well as rules about legal requirements, costs, manufacturing processes and other constraints.</td>
</tr>
<tr>
<td>Ullman, 1991, 1994</td>
<td>Have information about constraints and underlying assumptions (of the system) that are immediately accessible during the process of developing the design.</td>
</tr>
<tr>
<td>Candy &amp; Edmonds, 1996</td>
<td>Have access to historical and practical knowledge.</td>
</tr>
<tr>
<td>Maccoby, 1991</td>
<td>Draw upon own significant domain knowledge.</td>
</tr>
<tr>
<td>Cross &amp; Cross, 1996</td>
<td>Have access to means to extend own domain knowledge.</td>
</tr>
<tr>
<td>Roy, 1993</td>
<td>Have access to leading-edge events in the field. Keep in touch with other key players. Develop a network of contacts with active experts. Have access to technical literature for ideas applied in practice.</td>
</tr>
<tr>
<td>Chrisitaans, 1992</td>
<td>Acquire and apply different types of knowledge.</td>
</tr>
<tr>
<td>Visser, 1995</td>
<td>Use strategic knowledge for generation of creative designs. Alternate rapidly between thinking and information processing. Interact between old and new information. Integrate information in order to generate ideas. Enable knowledge development: e.g. progression from use of basic domain knowledge to strategic knowledge as designer learns.</td>
</tr>
<tr>
<td>Candy, 1996</td>
<td>Be opportunistic as a strategy for change. Be able to adapt or devise new methods quickly to meet new circumstances. Seize upon new ideas for pure experimentation's sake.</td>
</tr>
<tr>
<td>Lawson, 1993</td>
<td>Support individual designer strategies. Identify and maintain strategic knowledge. Use individual design knowledge as well as the shared knowledge of design practice. Incorporate both private and public design spaces. Address the mechanisms for the communication of that knowledge to other designers. Allow designers to share knowledge using shared representations.</td>
</tr>
</tbody>
</table>
References


331


