

Copyright is owned by the Author of the thesis. Permission is given for a copy to be downloaded by an individual for the purpose of research and private study only. The thesis may not be reproduced elsewhere without the permission of the Author.

A Multimedia Information System for the Support of Studies of Behaviour

A dissertation presented
in partial fulfilment of the requirements
for the degree of
Doctor of Philosophy in Information Systems
at Massey University

Eva Heinrich

1999

Abstract

The research presented in this thesis describes the conceptualisation of multimedia information systems to provide advanced support for the study of behaviour, the specification and implementation of such a system, called PAC, and the application of PAC in case studies in the fields of education and psychology.

Researchers in disciplines such as education, psychology and sociology are concerned with the study of behaviour and record behaviour on video or audio to capture it for subsequent analysis. A variety of quantitative and qualitative analysis methods are employed across the disciplines. Despite different emphases in the study of behaviour across the disciplines, a common set of core analysis steps can be established and identified as *description* of behaviour, *retrieval* of descriptions according to common characteristics, and *interpretation* of these descriptions. With the advances in multimedia computing, data of multiple media formats have become very accessible even on end user computer systems. This suggests the investigation into new analysis tools for multimedia data to support the analysis of behaviour recordings.

A conceptualisation for a new multimedia information system was developed which aimed at supporting the core analysis steps of description, retrieval and interpretation, providing means for the combined analysis of behaviour recordings of multiple media formats, and facilitating quantitative and qualitative interpretation techniques. This analysis support was complemented by the ideas of setting a single study into a domain context to form, over time, a knowledge basis for future studies and by the capture of study conclusions in multimedia format to allow rich, informative reporting of study results. The single most important conceptualisation for a new system was the design of the Flexible Structured Coding Language, FSCL. This coding language allows formulating rich and precise descriptions of behaviour in a very flexible way. Due to the structure of the coding language it is possible to correctly and completely retrieve the subject – verb – object relationships within the description sentences.

Based on the conceptualisation, a specification for a concrete advanced multimedia information system to support the study of behaviour was developed and a system was implemented accordingly. This system, called PAC, was used in three case studies concerned with the study of behaviour in collaboration with domain experts from the disciplines of education and psychology. The case studies aimed at testing the concepts behind the implementation PAC. By conducting their analyses using PAC, the domain experts gained detailed familiarity with the conceptualisation developed in this research and confirmed the usefulness of this conceptualisation for the study of behaviour.

Acknowledgements

There are a number of people I want to thank for their support throughout this research. My first thanks go to my two supervisors, Prof Jon Patrick and Dr Elizabeth Kemp. In particular, I am grateful to Prof Patrick who, through his own research, introduced me to the research area of information systems support for the study of behaviour and created an environment in which I could follow up on my own research ideas. I am especially thankful to Dr Kemp for her conscientious and thorough feedback throughout the research and especially over the period of writing up the thesis.

Many people from various disciplines have given me the opportunity to discuss the requirements of the disciplines involved in the study of behaviour and have listened to and commented on my evolving ideas for the design of a multimedia information system. I am especially thankful to the following researchers for conducting the case studies with me: Dr Alan Winton and Sasha McComb, Massey University; Dr Ian Wilkinson, University of Auckland; and Craig Whittington, Massey University.

On a personal note, I want to thank my parents for sending me off on the right path and for always being there for me should the need for support arise.

Publications

Publications related to this research are:

Heinrich E., Kemp E. and Patrick J.D. (1999). A Natural Language Like Description Language. *Tenth Australasian Conference on Information Systems (accepted)*.

Heinrich E., Patrick, J.D. and Kemp, E. (1999). *A Flexible Structured Language for the Coding of Human Behaviour: FSCL*. Technical Report 1/99, Department of Information Systems, Massey University, New Zealand.

Heinrich E., Patrick, J.D., and Kemp, E. (1998). A Multimedia Information System to Support the Analysis of Human Behaviour Recorded on Video. *Computers in Psychology*, Conference Proceedings, CTI Centre for Psychology, University of York, York.

Schroeter (Heinrich) E. and Patrick, J.D. (1996). An Application Interface with Multiple Workspaces: The Context Switcher. *Conference Proceedings OzCHI'96*. IEEE Computer Society Press, Los Alamitos.

Brief Contents

Chapter 1	Introduction	1
Chapter 2	Analysis Methods for the Study of Behaviour	11
Chapter 3	Existing Computer Systems to Support the Study of Behaviour	49
Chapter 4	Conceptualisation of Advanced Multimedia Information Systems	83
Chapter 5	The Flexible Structured Coding and Query Languages	101
Chapter 6	Towards a Specification for a Multimedia Information System	135
Chapter 7	Case Studies	155
Chapter 8	Conclusions and Further Work	201
 References		211
Appendix A	Summary of FSQL Query Options	223
Appendix B	Examples for FSQL Query Matching Algorithm	227
Appendix C	The PAC Windows Belonging to Each Module	235
Appendix D	The PAC Database Tables	239
Appendix E	Screenshots from the Implementation PAC	249

Detailed Contents

Chapter 1 Introduction	1
1.1 The Scope of ‘Studies of Behaviour’	1
1.2 Motivation and Objectives for the Research	2
1.3 Research Methodology	4
1.4 Structure of the Thesis	9
 Chapter 2 Analysis Methods for the Study of Behaviour	 11
2.1 Quantitative Analysis Methods	12
2.1.1 Study Design	12
2.1.2 Observer Effects	18
2.1.3 Validity and Reliability	20
2.1.4 Analysis and Interpretation of Data	22
2.1.5 Reporting of Results	25
2.2 Qualitative Analysis Methods	25
2.2.1 Qualitative Data	28
2.2.2 Analysis and Interpretation	30
2.2.3 Validity and Reliability	33
2.2.4 Reporting of Results	35
2.2.5 Use of Computers	36
2.3 Comparison of Quantitative and Qualitative Analysis Methods	38
2.3.1 Differences between Quantitative and Qualitative Analysis Methods	38
2.3.2 Commonalities of Quantitative and Qualitative Analysis Methods	42
2.4 Summary	45
 Chapter 3 Existing Computer Systems to Support the Study of Behaviour	 49
3.1 Existing Systems	50
3.1.1 OBSERVER	51
3.1.2 CABER	52
3.1.3 MacShapa	54
3.1.4 vPrism	55
3.1.5 Constellations	55
3.1.6 CVideo	56
3.1.7 NUDIST and CVideo	56
3.1.8 ATLAS.ti	58
3.1.9 Code-A-Text	59
3.1.10 FERAL	59
3.1.11 Other Systems	60

3.2	Discussion of Features of Existing Systems	65
3.2.1	Origin of the System	66
3.2.2	Focus on Analysis of Video Data	66
3.2.3	Use of Video Technology	67
3.2.4	Description Mechanisms	67
3.2.5	Retrieval of Information from Description Data	71
3.2.6	Storage of Analysis Data	73
3.2.7	Reporting on Results	73
3.3	Areas for Improvement	74
3.3.1	Technology	74
3.3.2	Multimedia	75
3.3.3	Combined Quantitative and Qualitative Analysis Systems	76
3.3.4	Coding Schemes	76
3.3.5	Database Management System	79
3.3.6	Interface Design	79
3.3.7	Support for the Whole Behaviour Analysis Process	80
3.4	Conclusion	80
Chapter 4 Conceptualisation of Advanced Multimedia Information Systems		83
4.1	Use of Multimedia Technology	85
4.2	Use of a Database Management System	89
4.3	Need for Flexible Structured Coding and Query Languages	91
4.3.1	Natural Language for the Description of Behaviour	91
4.3.2	Structured Languages for the Coding of Behaviour	92
4.3.3	Comparison of Natural Language Description and Formalised Coding	93
4.4	Interpretation Techniques	94
4.5	Specific Requirements of Quantitative Interpretation in a Multimedia Context	97
4.6	Support for Conclusion Construction	98
4.7	Domain Context and Knowledge Building	100
4.8	Conclusion	100
Chapter 5 The Flexible Structured Coding and Query Languages		101
5.1	The Flexible Structured Coding Language (FSCL)	101
5.1.1	Concept of FSCL	102
5.1.2	Categories of FSCL	102
5.1.3	Vocabulary of FSCL	104
5.1.4	Formal Specification of FSCL	104
5.1.4.1	Automaton and LL2 Grammar	106
5.1.4.2	Semantic Tree	107
5.1.5	Limitations of FSCL Compared to Natural Language	110
5.1.6	Coding Procedure	113
5.2	The Flexible Structured Query Language (FSQL)	114
5.2.1	Requirements of Correctness and Completeness	114
5.2.2	Flexibility in Querying	117
5.2.3	Combination of Code Instances through Querying	118

5.2.4	Format of the Query Results	126
5.2.5	Formal Description of FSQL	128
5.2.6	Query Evaluation Using FSQL	129
5.2.6.1	Construction of Full Query Tree	130
5.2.6.2	Application of Full Query Tree to Set of Code Instances	130
5.3	Conclusion	132
Chapter 6 Towards a Specification for a Multimedia Information System		135
6.1	Modular Design	136
6.2	Interaction between Modules and Database Tables	138
6.3	Interface Design using Module based Window Management	142
6.4	Use of Sets	144
6.5	The Implementation of PAC	148
Chapter 7 Case Studies		155
7.1	Case Study Design	155
7.1.1	Case Study Question	156
7.1.2	Multiple Case Design and Study Selection	156
7.1.3	Case Study Evidence	157
7.1.4	Analysis of Evidence	158
7.1.5	Cross Case Evaluation	159
7.2	The Individual Case Studies	160
7.2.1	The Reading Groups Case Study	161
7.2.1.1	Familiarisation with PAC and Scoping of Case Study	161
7.2.1.2	Process of Conducting the Case Study	162
7.2.1.3	Data Analysis using PAC	163
7.2.1.3.1	Behavioural Input Data	163
7.2.1.3.2	Development of Vocabulary	164
7.2.1.3.3	Coding	166
7.2.1.3.4	Querying of Code Instances	169
7.2.1.4	Shortcomings of the Implementation	170
7.2.1.5	Extensions and Changes to PAC Caused by the Case Study	170
7.2.1.6	Case Study Evaluation	172
7.2.2	The Dog Calming Case Study	173
7.2.2.1	Familiarisation with PAC and Scoping of Case Study	174
7.2.2.2	Process of Conducting the Case Study	175
7.2.2.3	Data Analysis using PAC	175
7.2.2.3.1	Behavioural Input Data	176
7.2.2.3.2	Development of Vocabulary	176
7.2.2.3.3	Coding	178
7.2.2.3.4	Querying and Interpretation	180
7.2.2.4	Shortcomings of the Implementation	181
7.2.2.5	Extensions and Changes to PAC Caused by the Case Study	182
7.2.2.6	Case Study Evaluation	183

7.2.3	The Parkinson's Rating Scheme Case Study	186
7.2.3.1	Familiarisation with PAC and Scoping of Case Study	186
7.2.3.2	Process of Conducting the Case Study	188
7.2.3.3	Data Analysis using PAC	188
7.2.3.3.1	Behavioural Input Data	188
7.2.3.3.2	Development of Vocabulary	189
7.2.3.3.3	Coding	190
7.2.3.3.4	Querying of Code Instances	192
7.2.3.4	Shortcomings of the Implementation	193
7.2.3.5	Extensions and Changes to PAC Caused by the Case Study	194
7.2.3.5	Case Study Evaluation	195
7.3	Cross Case Evaluation	197
7.4	Case Studies Summary	197
Chapter 8	Conclusions and Further Work	201
8.1	Review	201
8.2	Contributions	205
8.3	Further Work	207
References		211
Appendix A	Summary of FSQL Query Options	223
Appendix B	Examples for FSQL Query Matching Algorithm	227
Appendix C	The PAC Windows Belonging to Each Module	235
Appendix D	The PAC Database Tables	239
Appendix E	Screenshots from the Implementation PAC	249

Figures and Tables

Figures

Figure 1.1	Research steps	5
Figure 1.2	Interaction of research steps	6
Figure 2.1	Steps in the quantitative research process (Neuman, 1994)	13
Figure 4.1	Phases in the behaviour analysis process	86
Figure 4.2	Relationships of description, retrieval and interpretation techniques	95
Figure 5.1	Exemplar vocabulary definition	105
Figure 5.2	Automaton for first pass of syntax analysis with equivalent regular expression	108
Figure 5.3	Coding sentence example with pattern string and modified pattern string	109
Figure 5.4	Generic LL2 grammar for FSCL	110
Figure 5.5	Semantic tree for example sentence	111
Figure 5.6	Semantic trees for two example sentences	115
Figure 5.7	Query tree for example query	116
Figure 5.8	Query layers	120
Figure 5.9	Boolean OR query	121
Figure 5.10	Boolean AND query	122
Figure 5.11	Boolean NOT query	123
Figure 5.12	Time/Position comparison evaluation	124
Figure 5.13	Sequence query option	126
Figure 5.14	Combined-with query option	127
Figure 5.15	LL2 grammar for FSQL	129
Figure 5.16	Full query tree example	131
Figure 5.17	Query matching algorithm	133
Figure 6.1	The modules of PAC	138
Figure 6.2	Execution dependencies between modules and database tables	140
Figure 6.3	Result dependencies between modules and database tables	141
Figure 6.4	Module based interface elements in PAC	144
Figure 6.5	Role of sets in the analysis process	146
Figure 6.6	Potential coding - querying - coding sequence	147
Figure 6.7	Potential analysis sequence aimed at quantitative analysis	148
Figure 6.8	Potential analysis sequence aimed at qualitative analysis	149
Figure 6.9	Use of sets to combine quantitative and qualitative analysis	150

Figure 7.1	Questions to the domain experts	159
Figure 7.2	Specifications for digital video in Reading Groups case study	164
Figure 7.3	Behaviour recording example for Reading Groups case study	165
Figure 7.4	Abbreviated vocabulary in Reading Groups case study	167
Figure 7.5	Examples of coding sentences for word learning opportunities in Reading Groups case study	169
Figure 7.6	Specifications for digital video in Dog Calming case study	177
Figure 7.7	Behaviour recording example for Dog Calming case study	177
Figure 7.8	Vocabulary in Dog Calming case study	179
Figure 7.9	Examples of coding sentences in Dog Calming case study	180
Figure 7.10	Examples of external noise descriptions in Dog Calming case study	180
Figure 7.11	Initial queries in Dog Calming case study	181
Figure 7.12	Graphical presentation of correlating behaviours in Dog Calming case study	182
Figure 7.13	Specifications for digital video in Parkinson's Disease case study	189
Figure 7.14	Behaviour recording example for Parkinson's Disease case study	190
Figure 7.15	Vocabulary in Parkinson's Disease case study	191
Figure 7.16	Example coding sentences in Parkinson's Disease case study	192
Figure 7.17	Example annotations in Parkinson's Disease case study	192
Figure 7.18	Example queries in Parkinson's Disease case study	193
Figure 7.19	Example query results in Parkinson's Disease case study	194

Tables

Table 2.1	Comparison of quantitative and qualitative analysis	47
Table 4.1	Comparison of different approaches to description	94
Table 5.1	Transition table for automaton	109
Table 5.2	Vocabulary of two example sentences	115
Table 5.3	Flexible query options	119
Table 6.1	Association between PAC database tables and modules	139
Table 7.1	Propositions and main source of evidence	160
Table 7.2	Answers across case studies	198

Chapter 1

Introduction

1.1 The Scope of ‘Studies of Behaviour’

A wide range of disciplines is concerned with the study of behaviour. Hardyck and Petrinovich (1975) link the research on human behaviour to the fields of psychology, anthropology, sociology, education, and psychiatry. Rosnow and Rosenthal (1996) write that aspects of human behaviour are of interest to researchers in psychology, mass communication, education, sociology, anthropology, economy, psycholinguistics, behavioural biology, neuroscience, and statistics. According to Stangor (1998), behavioural research is conducted by scientists in fields like behavioural medicine, communications, criminology, human development, education, psychology, and sociology. Darst, Zakrajsek and Mancini (1989) write about the study of behaviour in the areas of physical education and sport .

Across the range of disciplines mentioned researchers focus on different aspects of behaviour. Stangor (1998, p. 3) describes the goal of behavioral research as “to discover, among other things, how people perceive their world, how they think and feel, how they change over time, how they make decisions, and how they interact with others.” Tuckman (1999, p. 400) writes that a qualitative researcher “should examine and report not only the behavior that took place but also the reasons or plans behind the behavior.” Rosnow and Rosenthal (1996, p. 12) state that “*behavioral science* is an ‘umbrella’ term that also includes cognitive functioning.” Writing on research methods in sociology McNeill (1985, p. 1) says that “sociologists are interested in those aspects of human behaviour that are the result of the social context in which we live.” Ray (1993, p. 19) distinguishes between the inner experience and the outer appearance, the behaviour, in studying psychological processes and states the importance of using “behavior to study subjective experience indirectly.”

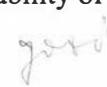
The research described in this thesis is concerned with conceptualising an information system to support the study of behaviour recorded in the form of multimedia data. The process of recording behaviour for subsequent analysis is performed in many different fields of study and suggested for example in (McBurney, 1994; Heath, 1997; Kent and Foster, 1977; Sommer and Sommer, 1991; Cozby, 1997; and Mars, 1989).

The following examples indicate the range of studies using behaviour recordings:

- Interaction analysis in workplace ethnography (Ruhleder and Jordan, 1998);
- Analysis of team sports (Patrick and McKenna, 1986a and 1986b);
- Psychotherapy research (Carter, 1997);
- Ethnographic Multimedia Narratives (Goldman-Segall, 1995);
- Eye-position recording (Stampe, 1993);
- Occupational therapy (McGeorge, 1987);
- Behaviour modification (Dowrick, 1991).

Considering the wide range of disciplines these studies come from and, therefore, the variety of quantitative and qualitative research methods applied within these studies, it is difficult to find one single term researchers from all disciplines can identify with. Throughout this research the term 'study of behaviour' has been used to refer to the wide range of studies mentioned here.

1.2 Motivation and Objectives for the Research

In 1982 Fassnacht wrote that the availability of video equipment had caused new interest in the observation of human behaviour (Fassnacht, 1982). This was followed by Hartmann and Wood stating in 1990 that computers were the most promising development in behaviour observation technology (Hartmann and Wood, 1990). It might be that now, in the late 1990's, the ready availability of digital multimedia technology can further advance the study of behaviour. 

The motivation for this research is to analyse from an information systems point of view what data analysis steps are undertaken in the quantitative and qualitative studies of behaviour and how these can be supported, across the disciplines the studies originate from, by employing information systems concepts. Knowledge from areas like databases, data structures, formal languages, human computer interaction and especially multimedia

is to be applied to conceptualise and implement an advanced multimedia information system to support the study of behaviour.

This motivation is to be supported by the following ideas:

- The variety of ‘studies of behaviour’ has, in the past, prompted the development of computer analysis tools originating from the three different directions of behavioural, cognitive and social traditions (Sanderson and Fisher, 1997). The idea behind this research is to conceptualise a generic analysis tool removed from the specific needs and view points of one behavioural domain. The focus of this generic analysis tool is to be able to support a wide range of studies of behaviour emphasising the commonalities across these studies. This goal is to be achieved by applying an information systems perspective, after careful study of the needs of the different behavioural domains and the existing solutions in form of computer applications.
- Many of the current computer systems which support the study of behaviour have been designed focusing mainly on data of one media type (examples are for video data: CABER, OBSERVER, vPrism; for text data: NUDIST, ATLAS.ti, Code-a-Text). Conceptualising a system as a multimedia system opens up a range of possibilities otherwise not available. The combined analysis of data of different media types can be facilitated. Ways can be investigated to provide analysis techniques for multimedia data commonly only available for textual data. New analysis techniques applicable to specific media types can be designed.
- Designing a multimedia system implies that the data of the different media types are all available within the system which can lead to an extended support for a study of behaviour. Multimedia analysis data collected during the course of the study can be used to support and illustrate the findings of the study in form of a multimedia presentation. These multimedia presentations can facilitate the exchange of knowledge across several studies belonging to a common research theme.

Following from the motivation the main objectives of this research are:

1. to examine the quantitative and qualitative analysis methods applied and the computer tools used to support studies of behaviour;
2. to identify areas where the use of information technology and specifically multimedia can lead to the development of an advanced information system to support studies of behaviour;

3. to conceptualise, design and build such an information system;
4. to evaluate the new ideas by using the implementation to conduct selected studies of behaviour in conjunction with behavioural domain experts.

1.3 Research Methodology

Galliers (1994, p. 159) has developed a revised taxonomy of information systems research approaches and suggests that this taxonomy can be regarded as a framework for selecting the most suitable research approach. Based on the taxonomy, a descriptive/interpretive approach was taken to investigate the methodologies employed in the studies of behaviour and the technology, that is the existing computer systems to support the study of behaviour. These research steps were performed to ensure that the following step of theory building would be founded on existing knowledge and would consider the requirements of the field of study of behaviour. The theory building was again performed using a descriptive/interpretive research approach in conceptualising a new information system to support the study of behaviour. For the theory testing two research approaches were applied to facilitate the testing of the theories from different perspectives. A prototype application was developed (a research approach included in Galliers taxonomy under the heading of 'Theorem proof') and case studies were performed. The theory extension was based on both research steps in theory building, prototype development and case studies. Figure 1.1 presents an overview of the research steps.

While these steps were largely undertaken in sequence as shown in Figure 1.1 there was as well a certain degree of overlap between the steps of theory building and theory testing. Insights gained during the prototype development highlighted conceptual problems in the theory which had to be rectified for the development to continue. The work on the case studies, which were conducted over a timeframe of around one year, had an impact on both the theory building and the theory testing through development. In some instances discussions with the domain experts in the case studies revealed desirable features which were not implemented in the prototype. Where feasible and of value to the case study these features were added to the implementation. Some features suggested by the case studies had not only impact on the implementation but as well on the concepts behind the implementation. In these few instances it was necessary to go back from the

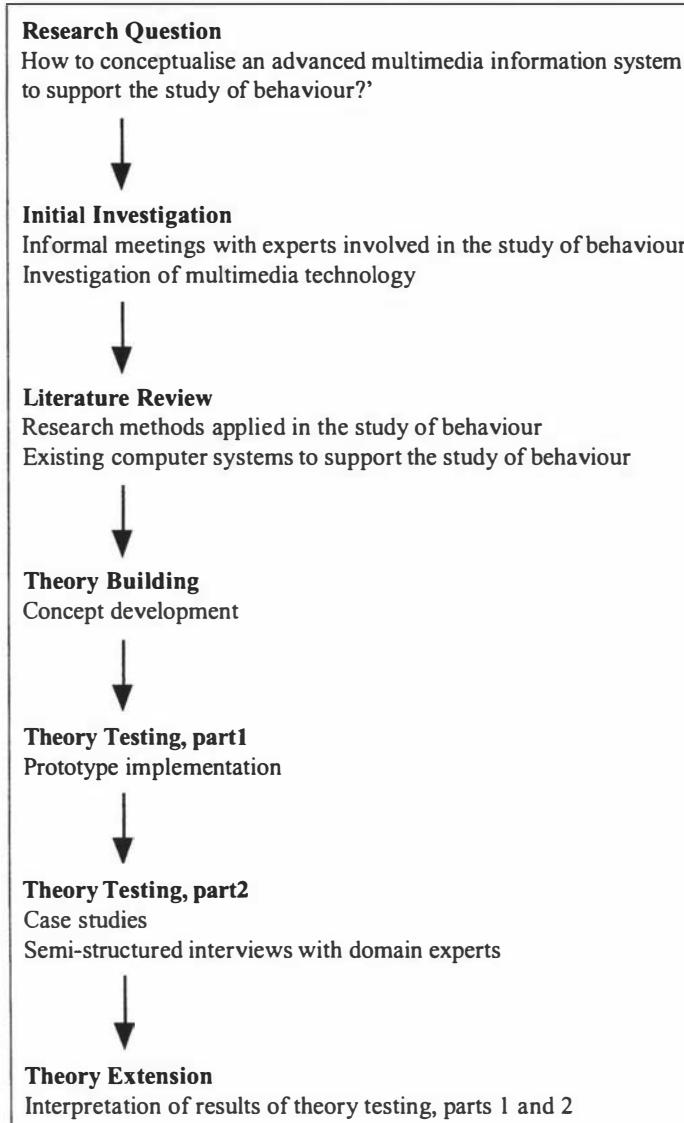


Figure 1.1: Research steps

case studies to theory development and from there to implementation. Figure 1.2 indicates the connection of these research steps along a timeline.

Research Question

The research question asked was ‘how to conceptualise an advanced multimedia information system to support a wide range of studies of behaviour?’. The motivation for this research question is outlined in detail in Section 1.2.

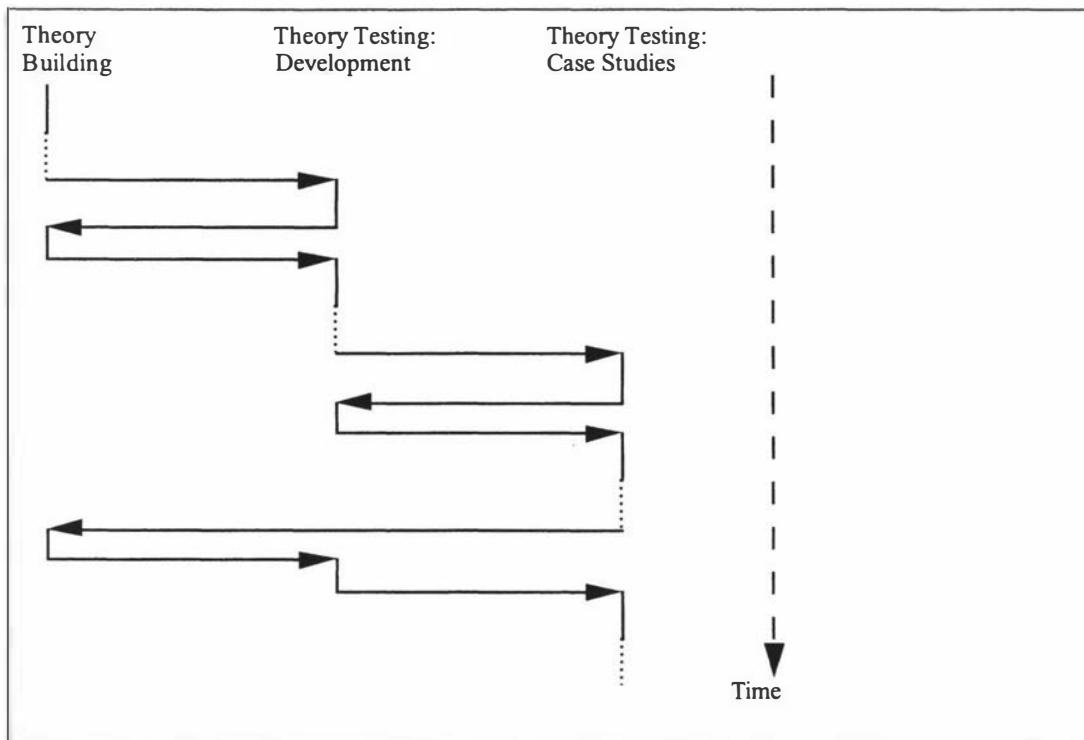


Figure 1.2: Interaction of research steps

Initial Investigation

Initial investigations were performed to develop an understanding of different types of studies of behaviour and the advances in multimedia technology. Many informal meetings with experts working in different areas of study of behaviour were arranged. These meetings served a dual purpose. Firstly, the characteristics of the different behavioural domains and the analysis methods employed were collected. Secondly, the meetings presented a chance for asking the experts about their requirements for a computer system to support their analysis work. The experts were researchers or professionals from a range of different backgrounds:

- A group of medical staff from the local hospital working in a section for the care of elderly patients;
- A researcher in psychology who had an extensive research background in analysing the development of children;
- A researcher in psychology involved in conversation analysis applying qualitative research methods;

- A researcher working in both areas of information systems development and the training of psychodrama directors;
- A researcher focusing on behaviour modification applying quantitative research methods;
- A researcher in sports management and coaching;
- A national level sports coach;
- A group of researchers working in media studies applying both quantitative and qualitative research methods;
- A researcher applying knowledge elicitation techniques to investigate computer interface issues;
- A researcher with experience in studying the interaction in team sports;
- A researcher in human computer interaction;
- A researcher in behaviour modification using video recordings to facilitate behaviour changes.

The technical feasibility of working with digital media, mainly video recordings, was analysed in preliminary tests. Behaviour recordings were digitised using available tools to develop an understanding of file sizes, conversion times, replay performance, and the ratio of file sizes to image quality. Tools were developed to test the feasibility of first conceptual ideas like the provision of graphical annotation tools for the video recordings. The outcomes of the initial tests using digital media indicated that it would be possible, with the equipment available at that point of time, to develop a multimedia information system based on the use of digital media. While it was obvious that some performance aspects of dealing with digital video were not fully satisfactory, it could be assumed that the rapid technical advances in the area of multimedia computing would help to overcome these limitations in a matter of very few years. It was judged a prototype implementation would be able to handle multimedia data sufficiently well to allow the testing of the conceptual dimensions of the research.

Literature Review

The literature review phase of the research looks at the research methods employed in the study of behaviour and the existing computer programs to support the study of behaviour.

To conceptualise tools for the support of the study of behaviour it was essential to investigate the research methods used in these studies. Only with an intimate knowledge of these methods could it be possible to propose an information system to support the study of behaviour.

A number of computer programs exist to support the study of behaviour. These programs represent a variety of behaviour analysis domains from different research disciplines. It was important for this research to evaluate what solutions to support the study of behaviour had been proposed and implemented. This evaluation was to give an insight into the strength and weaknesses of current programs and was to lead to the identification of areas for improvement.

Theory Building

Based on the literature review concepts for a multimedia system to support the study of behaviour were developed. The needs of the user domain, the current solutions available to that domain, and the possibilities arising from the technological advances of multimedia computing were identified. Information systems knowledge from areas like data structures, databases, formal languages, user interface design and systems development was applied to conceptualise and develop an advanced multimedia information system.

Theory Testing, Part1

A detailed specification for the new concepts was developed to allow the implementation of a prototype application. The systems development phase was seen as an essential step in the overall research design. Developing the prototype application made it possible to demonstrate the feasibility of the concept and to conduct the next step in the research process, the observation via case studies. The use of software development to complement other information systems research strategies is, for example, suggested in Nunamaker, Chen and Purdin (1990/91) and Parker, Wafula, Swatman and Swatman (1994).

Theory Testing, Part2

This research phase was conducted in the form of case studies and semi-structured interviews. The case studies were conducted in collaboration with domain experts from

different behavioural domains. The usefulness of case studies as information systems research strategy is described in Avison and Nandhakumar (1995) and Yin (1993).

Theory Extension

The insights gained in the two phases of theory testing were interpreted. The strength and weaknesses of both conceptualisation and implementation of the multimedia information system to support the study of behaviour were evaluated.

1.4 Structure of the Thesis

The structure of the thesis follows closely the steps in the research methodology. Chapter 2 contains the review of quantitative and qualitative research methods in the disciplines of education, psychology and sociology. In Chapter 3 existing systems to support the study of behaviour are reviewed. The new conceptualisation of multimedia information systems to support the study of behaviour is described in Chapter 4, with Chapter 5 concentrating on the new coding and query languages. The specification for a specific system, called ‘PAC’, is outlined in Chapter 6. Chapter 7 contains the report on the case studies and on the semi-structured interviews with the domain experts. Chapter 8 on review and further research directions concludes the thesis.

Chapter 2

Analysis Methods for the Study of Behaviour

In this chapter the literature on analysis methods of disciplines concerned with the study of behaviour is reviewed. As indicated in Chapter 1, a wide range of disciplines are concerned with the study of behaviour. The literature review presented here concentrates on the behaviour analysis methods applied in the areas of education, psychology, and sociology. The review of these three areas, for the purpose of this thesis, covers a sufficiently wide range of disciplines concerned with the study of behaviour.

Beside the range of disciplines studied, the review of the analysis methods was conducted in the light of constructing a multimedia information system to support the study of behaviour. Such a multimedia system would, in the first instance, rely on the capture of behaviour in the form of video, audio, or image recordings for the subsequent analysis. In the second instance the multimedia system could provide tools to analyse descriptions of behaviour in the form of text documents, transcripts of audio recordings, or additional descriptive material. Because the literature review was to serve as background for the conceptualisation of an information system, the analysis methods had to be studied on a fairly detailed level, concentrating less on theories than on particular analysis steps undertaken.

A literature review shows that a number of authors (Neuman, 1994; Burns, 1997; Krathwohl, 1998; Gay, 1996) divide the analysis methods into quantitative and qualitative methods. Sections 2.1 and 2.2 of this chapter follow this outline. To prepare a later argument for conceptualising a multimedia system which supports both quantitative and qualitative analysis methods Section 2.3 examines the literature discussing the benefits of such a combined analysis. Section 2.4 concludes this chapter by summarising the review of the analysis methods and highlighting the commonalities and differences of the quantitative and qualitative approaches.

2.1 Quantitative Analysis Methods

Quantitative research in general is concerned with the collection of numerical data for testing of hypotheses in order to explore wider theories. It uses statistical methods to analyse data, which are collected according to a closely specified concept developed during the initial phases of the research process (Neuman, 1994). The quantitative research process, as defined for example in Neuman (1994) and pictured in Figure 2.1, contains a number of steps from selection and refinement of the research question, study design, data collection, data analysis and interpretation to reporting on the study findings. The quantitative data analysis has to be seen in context of the research process (Bryman and Cramer, 1994). Earlier steps in the research process, like study design and data collection, impact on the quantitative analysis steps. While the quantitative research process is presented mainly as a series of linear steps, it can be necessary to revisit earlier steps in the study process (Bordens and Abbott, 1996). Research questions lead to the definition of measurements. Measurements produce data. The data are analysed and these analysis results will lead to the formulation of new research questions. Preliminary observations interact with the formulation of research questions and hypotheses, the definition of measurements and recording methods (Martin and Bateson, 1993). In a wider context, looking at a number of studies in a domain area, results of one study cycle back into the preparation of another study (Bordens and Abbott, 1996; Martin and Bateson, 1993).

In the light of providing the foundation for the development of an information system which supports the quantitative analysis of behavioural data, a number of steps of the quantitative research process have to be reviewed. As part of the study design, observation methods, operational definitions, measures of behaviour and recording methods have to be discussed. The observer effects on the data collection have to be considered. The validity and reliability of the study will be decided by the study design and the way the data are collected. The data have to be analysed and interpreted and finally the findings of the study have to be reported.

2.1.1 Study Design

Bentzen (1997) describes some general characteristics of formal behaviour observation methods. He distinguishes between open and closed methods. Data can be collected with

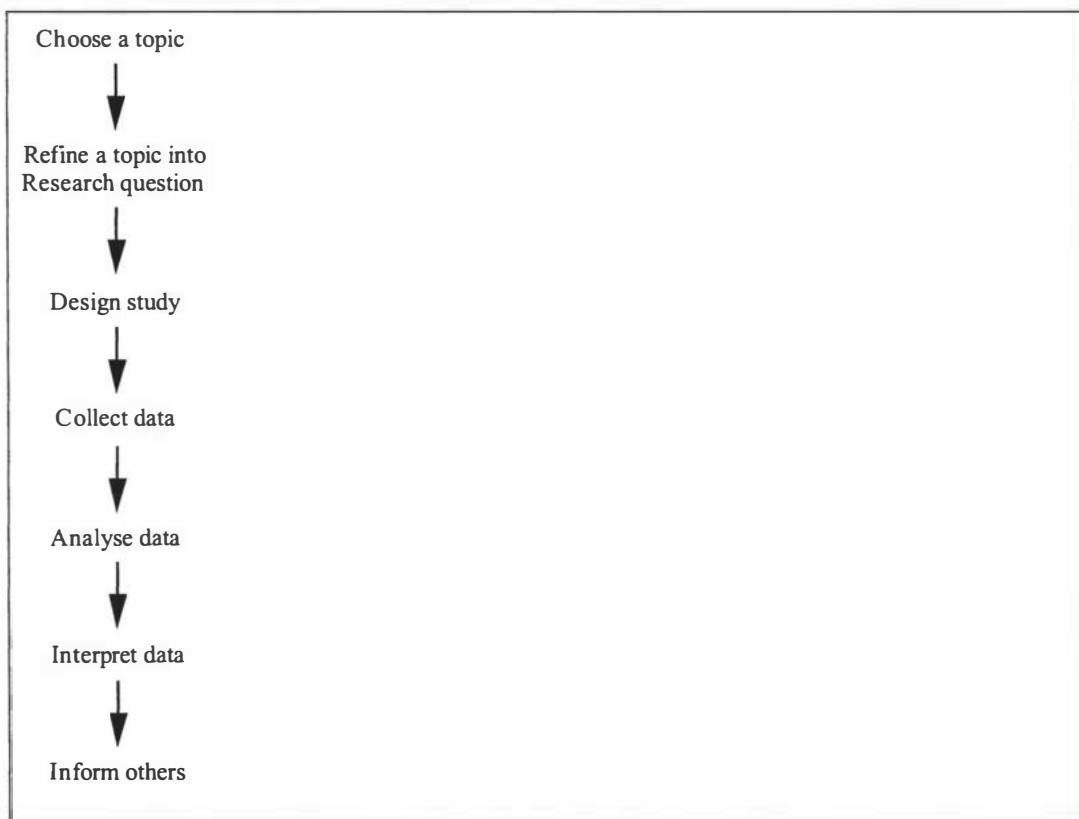


Figure 2.1: Steps in the quantitative research process (Neuman, 1994)

open methods resulting in raw data. Raw data are descriptions of behaviours which capture the behaviours as they originally occurred. These raw data can be looked at at any point in time to extract quantitative data. Often in behaviour analysis closed methods are used. Data collected using closed methods have already undergone interpretation, as during observation, behavioural events have been converted into summaries and counts. Closed methods do not capture the behaviour in its full description, they extract data agreed on operational definitions. The question of using open or closed methods for observation is linked to the degree of selectivity. In unselective data collection no specific behaviours are targeted for observation and recording. In selective data collection the behaviours to be recorded are specified in detail before the observations take place. A further consideration in the design of the observation method has to be the degree of observer inference. Inferences or interpretations produce conclusions which are only based on observations but not directly observable.

Before the data collection can be conducted a number of criteria have to be considered. Operational definitions specify exactly what aspects of behaviour are to be measured while the measures of behaviour and the recording methods define how the data are to be collected.

Operational Definitions

Operational definitions state how a conceptual variable, expressing an abstract construct like depression or self-esteem, can be linked to a measured variable, standing for a measurable form of behaviour like distance in meters or a number of instances (Stangor, 1998). Tuckman (1999, p. 113) describes an operational definition as a “characterization based on the observable traits of the object being defined” and stresses the significance of the word observable. Detailed operational definitions support the understanding of how observations have been arrived at and enable the replication of research (Stangor, 1998). Within one study precise operational definitions are necessary to ensure consistency among the observers (Hartmann and Wood, 1990). Often the definitions are tested in pilot observations prior to the start of the data collection. The experiences gained in the pilot phase are used to refine the operational definitions (Hartmann and Wood, 1990).

Operational definitions consist of a number of components. According to Hartmann and Wood (1990) operational definitions should refer to directly observable components of the target behaviour. The operational definitions should be clear and unambiguous. Beside a descriptive name, the operational definitions should include further information to make their intended application obvious to allow for consistent use by several observers through a range of settings. This additional information should include a detailed description of the critical parts of the behaviour, typical examples of the behaviour and questionable instances of the behaviour. This information can be provided as textual description, supported by images displaying the critical aspects of the behaviour to be measured.

Hartmann and Wood (1990) go on to distinguish between molar and molecular operational definitions. Operational definitions can, depending on the needs of a particular study, be defined in different levels of breadth and precision. Molar operational definitions are used to measure global units of behaviour, whereas molecular operational definitions are used to code more narrowly defined units of behaviour. Because molecular

operational definitions refer to more specific aspects of behaviour than molar operational definitions they are in general easier to apply. An additional advantage of molecular operational definitions is that they can be collapsed into molar operational definitions to provide data for summary analysis.

Within the observation system operational definitions can be used in an exclusive and/or exhaustive way and have to be defined accordingly (Hartmann and Wood, 1990). Exclusive use of operational definitions means that for each unit of time only one act or series of acts of behaviour can be recorded. Priority coding rules have to be established to determine which behaviour is recorded and which behaviour is ignored if behaviours occur simultaneously in one unit of observation. Alternatively, the system of operational definitions can provide specific operational definitions for all possible combinations of behaviours occurring simultaneously. If the observation system is defined to cater for exhaustive measuring of behaviour, one behaviour has to be recorded in each observational unit. The operational definitions have to be defined accordingly, so that suitable operational definitions are available for each unit.

Measures of Behaviour

The main types of measurement in behaviour analysis as described in Martin and Bateson (1993) are latency, frequency, duration and intensity. Latency is measured in units of time. It is defined as the time from a specified event to the start of the first occurrence of the behavioural event to be observed. Frequency is the number of occurrences of the behaviour pattern per unit of time. It is sometimes referred to as the rate of occurrence. Duration is measured in units of time. It is the length of time for which a single occurrence of the behaviour pattern lasts. Intensity cannot simply be measured by the absence or presence of a behaviour, as it is possible for the measures of latency, frequency and duration. Intensity has no universal definition, a measure of intensity has to be derived depending on the behaviour to be measured. One definition for intensity is the local rate, which is defined as the number of displays of a certain component of a behaviour during the length of the behaviour.

A further type of measurement is quality which can be assessed only indirectly. Often the quality of a behaviour can be expressed by rating the different levels of the behaviour and recording the occurrences of the behaviour in the different levels (Martin and Pear, 1992).

The measures most often used are frequency and duration. As one of these measures cannot be derived from the others, both should be measured to achieve a good description of the behaviour. Behaviours of relatively short duration, called events, are best measured as frequency of occurrence. Behaviours of relatively long duration, called states, are best measured via their duration (Martin and Bateson, 1993).

The reporting on these measurements should put the results in context with the observation intervals. A count of observed events should be set in context with the length of the observation. From the measurements of duration the percentage of the total observation time, in which the behaviour occurred, can be calculated (Martin and Bateson, 1993).

Recording Methods

To specify the recording method, sampling and recording rules have to be defined. The sampling rule specifies the subject of the observation and the time period whereas the recording rule decides how the behaviour is to be recorded (Martin and Bateson, 1993).

The sampling rules are divided into ad libitum, focal, scan, and behaviour sampling. Ad libitum sampling is a non systematic form of sampling which does not constrain what is recorded or when. This sampling method can be of use for preliminary observations in an exploratory phase of a study. Due to its non systematic nature it is of limited use in a structured investigation (Hartmann and Wood, 1990). Focal sampling implies the observation of one group of subjects for a specified length of time. All instances of behaviour displayed by the members of the group are recorded according to different categories of behaviour (Martin and Bateson, 1993). In scan sampling the behaviour of a group of subjects is scanned at regular intervals. For a number of categories of behaviour the behaviour is recorded if occurring at that exact instance of time (Hartmann and Wood, 1990). The behaviour sampling focuses on a specific type of behaviour. The whole group of subjects is observed and every occurrence of the behaviour is recorded in detail (Martin and Bateson, 1993).

The literature defines three recording rules. Martin and Bateson (1993) write about continuous recording and time sampling which they further divide into instantaneous sampling and one-zero sampling. Martin and Pear (1992) define continuous recording,

time sampling recording (equivalent to Martin and Bateson's instantaneous sampling) and interval recording (equivalent to Martin and Bateson's one-zero sampling). Continuous recording attempts to produce an accurate, complete record of the behaviour displayed. Every instance of behaviour is recorded with time of occurrence for events or start and end times for states. Continuous recording enables the calculation of true frequencies, latencies, and durations. In practise it can be difficult for the observer to perform continuous recording and it might be necessary to restrict the observations to a limited number of behavioural categories. Continuous recording is typically used for recording frequencies of events or durations of behavioural states. It is of particular importance to preserve information about the sequence of behaviour patterns (Martin and Bateson, 1993). In instantaneous sampling the observation session is divided into sample intervals. At the end of each sample interval, called the sample point, the presence or absence of a behaviour at that instance of time is recorded. This form of sampling can only provide an estimate of frequencies or durations. It is suitable to record behavioural states for which the occurrence or non occurrence can be decided at any point-of-time. It is not suitable for recording discrete events of short duration or for recording rare behaviour patterns as these are likely to fall in between the sample points. The accuracy of the instantaneous sampling is influenced by the length of the sample intervals and the average duration of the behaviour patterns (Martin and Bateson, 1993). The one-zero sampling is similar to the instantaneous sampling in dividing the observation session into intervals. In contrast, a behaviour is recorded depending on its occurrence at some stage during the time interval and not at a sample point. Like the instantaneous sampling this form of sampling can only provide estimates (Martin and Bateson, 1993).

Data Collection Tools

Depending on the design of the data collection method a range of tools is available to record the behaviour or specific aspects of the behaviour (Martin and Bateson, 1993). These tools can be distinguished both in terms of technology used and in the degree of analysis performed while recording the behaviour. Video or audio recording devices can be used for the continuous recording of behaviour, delaying the analysis of the behaviour largely to a later point in time. Verbal descriptions of behaviour can be captured as spoken words with an audio recording device or as written words, typed into a computer or hand written with pen and paper. If a coding scheme is involved and only certain

categories of behaviours are to be recorded, check sheets or computerised event recorders can be used. Check sheets are a pen and paper data recording method. They contain a matrix listing the behaviour categories and the sampling intervals. In accordance with the sampling method selected for a study an observer uses the check sheet to code the occurrence or form of a behaviour. With a computerised event recorder, pen and paper is replaced by a computing device. Behavioural events are entered via a keyboard (or a similar input device) and stored with the help of a specialised event recording software. The recorded data can then be analysed with the computer.

The selection of a data collection tool is influenced by a number of factors. Beside availability of the technology and a good match with the data collection method, issues like reliability, portability, obtrusiveness, or the availability of suitable observers have to be considered (Hartmann and Wood, 1990). The use of video recordings to capture behaviour has great potential (Agnew and Pyke. 1978). Video recordings are especially valuable where complex behaviours or rapidly changing behaviours have to be observed (Hartmann and Wood, 1990). The use of computers promises both advances in the recording of behaviour and in the analysis of the behaviour recordings (Hartmann and Wood, 1990).

2.1.2 Observer Effects

The quality of the data produced by the observers is dependent on a number of factors, ranging from the suitable preparation of the observation system to the observation environment and the training given to the observers. Gambrill (1977) states as the two most common reasons for inaccurate data the failures to specify behaviour categories clearly and to provide easy recording procedures.

To be able to produce accurate observations the observers have to be given suitable operational definitions and observation tools (Martin and Pear, 1992). If the operational definitions describe behaviour which is difficult to detect or if it is difficult to determine if the behaviour has occurred or not, the observers will have problems capturing accurate measures of the behaviour. In this case the behaviour to be recorded should be split into more easily recognisable sequences of behaviour. Tools like data sheets might have to be redesigned if they do not support the observers efficiently in their recording task. Additionally, the general environment has to be supportive for the recording task. A very

disruptive environment for example will it make difficult for the observers to concentrate on the task.

The study design has to consider the potential effects of the observations on the observed individuals (Hartmann and Wood, 1990). The presence of human observers and the awareness of being observed is likely to have an effect on the outcome of the observations. The observed individuals might react to being observed with changes to their normal, unobserved behaviour. The suppression of socially undesirable behaviour or the facilitation of socially desirable behaviour is called the valence of behaviour. The subject characteristics of individuals result in differently strong effects of the observation on the behaviour of the individuals. The effects of the observation on the displayed behaviour is also influenced by the conspicuousness of the observations, which is determined by how obtrusive or obvious the assessment procedures are. Further, the personality of the observer and the relationship of the observer to the observed individuals can influence the behaviour of the individuals and therefore the observation results. A thorough rationale for the observation procedures should be provided for the observed individuals in order to reduce their potential concerns and possible reactive effects due to the observation process.

Beside the mentioned effects of the design of the observation system and the influences on the displayed behaviour which can occur due to the fact that the behaviour is observed, the observers themselves can actively influence the observation results. Systematic errors in assessment can be caused by observer bias which is usually associated with the expectancies and prejudices of the observers and their limitations in processing information (Hartmann and Wood, 1990). The observation results can be affected by measurement decay in observer performance, called observer drift. The consistency and accuracy of an observer may change during the process of the study, as for example from the training phase to the phase of formal data collection. Further the observers can influence the observation results by intentional or unintentional fabrication, which should be monitored and prevented through random reliability checks. To increase the consistency of the observations, Gambrill (1977) suggests performing a thorough initial training of the observers followed by periodic retraining, emphasis on preventing observer fatigue and the collection of accurate data.

Hartmann and Wood (1990) outline a training schedule for the observers:

- Orientation of the observers;
- Learning of the observation manual: Learning of the operational definitions and measurement procedures;
- First criterion check: The observers should have working knowledge of the observation system;
- Analogue observations: Training to criterion accuracy and consistency; in this phase it is particularly useful to get the observers to measure behaviour recorded on video or film; alternatively interactive computer systems based on training videos can be used;
- In situ practice: Accustoming the observers with the real situation;
- Retraining calibration sessions: Testing the mastery of the observation manual; comparing observer results.

To reduce the possibilities of observer bias, the observers should not know the study hypotheses during the investigations. After the observations are finished, the observers should be informed about the nature and results of the study they participated in (Hartmann and Wood, 1990).

Closely related to the observation effects are the questions of validity and reliability of observations. These questions will be discussed in the next section.

2.1.3 Validity and Reliability

The usefulness of analysis results depends on the reliability and the validity of the measures derived from observations of behaviour. Reliability can be seen as the question of 'how good the measures are' or how precisely and consistently they measure a variable, whereas validity is the question of 'how right the measures are' or how well they actually answer the question being asked (Martin and Bateson, 1993). Both Stangor (1998) and Rosnow and Rosenthal (1996) introduce the concepts of validity and reliability by outlining the effects of *random* and *systematic errors* on the accuracy of measurements. A random error, also called chance fluctuation, tends to produce measurements fluctuating around the exact measurement value. Over repeated measurements random errors are likely to compensate each other resulting in an average measurement which is close to the accurate value. The random errors are therefore a threat to consistency or stability and are as such linked to the term reliability. Systematic errors, also called bias, tend to influence

a measure into just one direction and do not eliminate each other over a series of measurements.

Measurements in behavioural research have to be both reliable and valid. A measurement that is not reliable is unlikely to be valid, yet a non valid measurement can be very reliable (Rosnow and Rosenthal, 1996). The following paragraphs look in more detail at different forms of reliability and validity.

The reliability of a measurement can be tested by performing the same measurement several times. Assuming the parameters of the measurement have not changed one would expect to arrive at the same measures. The measurement could then be considered as reliable (Stangor, 1998). Different forms of reliability are test-retest reliability, internal consistency reliability, and interrater reliability (Stangor, 1998, Rosnow and Rosenthal, 1996). To check for test-retest reliability the same measurement is applied to the same group of subjects twice. A correlation coefficient can be calculated which expresses the degree of test-retest reliability (Rosnow and Rosenthal, 1996). Care has to be taken to consider retesting effects or the measuring of conceptual variables which can be expected to fluctuate for a given individual (Stangor, 1998). The internal consistency reliability looks at the degree of relatedness between individual measurements which refer to the same conceptual variable. Based on the correlation between individual measurements the overall internal consistency can be calculated (Stangor, 1998, Rosnow and Rosenthal, 1996). The internal consistency of observers, the intra-observer reliability, can be tested by having one observer measure the same behaviour twice, which can be achieved by videotaping the behaviour to be observed. Regular tests for intra-observer reliability should be integrated into the study design (Martin and Bateson, 1993). The interrater reliability (or inter-observer reliability) investigates the internal consistency of a group of observers. Measuring the same behaviour using the same set of instructions it has to be expected that two observers arrive at very similar measurements. If this is not the case the interrater reliability is low and it has to be investigated why the observers rate the same behaviours differently (Stangor, 1998; Martin and Bateson, 1993).

Cozby (1997) writes about three major forms of validity. Construct validity says to what extent a measurement reflects the conceptual meaning of a variable. Internal validity

refers to the ability to infer causal relationships between the variables of a study. External validity is determined by the extent to which the findings of a study can be generalised.

Construct validity can be seen as a framework for other forms of validity (Krathwohl, 1998). Face validity examines subjectively if a measurement is likely to measure what it seems to measure. A measure can appear to be appropriate to test a conceptual variable yet might, for example, be unlikely to elicit honest answers from individuals. A high face validity is not always desirable and in some circumstances a low face validity might contribute to a higher construct validity (Stangor, 1998). Content validity is concerned with the question of whether the measured variables relate to a sufficient range of aspects of the behaviour studied. The content validity is low if the measurements or tests only relate to a small subset of aspects (Rosnow and Rosenthal, 1996; Stangor, 1998). Different forms of criterion validity test the correlation of the measurements with behavioural measures external to the current test or investigation. Concurrent validity looks at correlations at the time of the investigation while predictive validity attempts to assess the prediction of events in the future (Stangor, 1998).

2.1.4 Analysis and Interpretation of Data

The analysis and interpretation of the behavioural data follow two main directions. These are the exploratory data analysis and the confirmatory data analysis. Exploratory data analysis is a process of searching through the data, of sorting and summarising the data. Results are often presented in graphical or other visual form. Exploratory data analysis aims at understanding the data and at creating new hypotheses from the data. It is performed with techniques provided by descriptive statistics. Confirmatory data analysis deals with the testing of hypotheses. It applies inferential statistics techniques. Confirmatory data analysis looks at what inferences can be made from the data and at the confidence with which these inferences can be made (Martin and Bateson, 1993). A hypothesis can be defined as a “tentative explanation for a phenomenon” (Dunham, 1988, p. 85). The creation of hypotheses can be described as an inductive process where ideas or theories arises from the analysis and interpretation of observations. The testing of hypotheses is a deductive process where well established theories predict observations (Ray, 1993). The analysis and interpretation using descriptive and interpretive statistics is

commonly performed with the help of statistical computer programs (Cozby, 1997; Ray, 1993).

The data available for analysis result from the measurement of variables, as described by a number of authors (Cozby, 1997; Ray, 1993; Krathwohl, 1998; Gay, 1996). Which statistical procedures can be applied sensibly to the data depends on their type of measurement scales. The four types of measurement scales are nominal, ordinal, interval, and ratio. A nominal scale has no numerical or quantitative property. A nominal variable assigns a name based on membership to a specific category or a specific property. Nominal scales can not be used for comparisons but rather for counting the members of a category and calculating, for example, frequency distributions. An ordinal scale introduces a quantitative distinction. It contains a ranking of members of a group along a continuum. An ordinal scale records only the order of values without making a statement about the distances between these values. An ordinal scale allows the comparison of members of a group. In an interval scale the distances between the numbers on the scale becomes meaningful. The distances between two neighbouring numbers on the scale are all the same. The differences of two values on an interval scale can be compared. Interval scales have an arbitrary reference point which distinguishes them from ratio scales. Ratio scales are like interval scales but additionally have an absolute zero point which indicates the absence of a property being measured. Ratio scales allow direct comparisons of two scales values resulting in statements of the form 'twice as fast/long/..'.

The first step in analysing data is often to construct a frequency distribution. A frequency distribution shows in a graphical or tabular format the frequency of scores received for a variable (Cozby, 1997). Graphs and tables are very suitable forms to present and summarise data and to illustrate the relationships between variables (McBurney, 1994; Rosnow and Rosenthal, 1996). Two very common graph forms for frequency distributions are frequency polygons and histograms. To display the relationship between two variables, tables, scattergrams, line graphs, or bar graphs are used (McBurney, 1994; Cozby, 1997).

Descriptive statistics play a number of roles in the analysis process. They are used to address specific questions stemming from the research design of the study, they facilitate the exploration of the data to discover new patterns and relationships, and they can test

the suitability of the data for the application of inferential statistical techniques (Bordens and Abbott, 1996). A number of authors describe the techniques of descriptive statistics (Ray, 1993; Rosnow and Rosenthal, 1996; Krathwohl, 1997; Gay, 1996). The following summary follows the presentation given in Ray (1993). The techniques of descriptive statistics assist in calculating measures of tendency, measures of variability, the transformation of data, and measures of association. The measure of tendency is expressed by the three values of mean, median, and mode. All these values represent some form of medium value across a population and it depends on the research question which value is the most appropriate to use. Of specific importance are normally distributed data which are represented by a normal curve. For these data, among other characteristics, mean, median, and mode values are the same. While the measure of tendency only provides information summarising a whole group of scores it does not show how the scores are distributed. This information is provided by the measures of variability. The range measure give the difference between largest and smallest score in a set of data. The variance describes how much each score varies from the mean value and leads to the calculation of the standard deviation which allows the making of meaningful comparisons between different sets of data. Linear transformations can be performed to change the measurement unit of values. Applied to all data they have no effect on the meaningfulness of the data. Another form of transformation is the calculation of standard and z scores. These scores put an individual score in relationship to the mean and standard deviation of the group of scores and allow for the comparison of scores of different sets of values. The measure of association gives information about the relationships of two measures. A correlation expresses the degree of association between the measures. Various correlation coefficients, like the Pearson coefficient, can be calculated.

Information about inferential statistics is provided by many authors (Bordens and Abbott, 1996; Cozby, 1997; Rosnow and Rosenthal, 1996; Gay, 1996; Krathwohl, 1997; Ray, 1993). The brief summary presented here will again follow the presentation of Ray (1993). The techniques of inferential statistics are used to make statements about a sample of measurement values in relationship to the whole population this sample was taken from. Questions like ‘are the results achieved for a specific sample representative for the whole population?’ are asked. To be able to apply inferential statistics some conditions

like random or systematic sampling have to be satisfied. Techniques of inferential statistics are used for hypothesis testing. A null hypothesis is formulated and it is then determined if and with what probability the null hypothesis can be accepted or rejected. A null hypothesis is often used to test for the effect of a particular treatment. Two groups are randomly sampled from a population and exposed to different treatments. The null hypothesis is formulated to state that there will be no difference for the two groups in performing a test. With techniques from inferential statistics, like the t-test, the correctness of the null hypothesis can be determined and it can be calculated with what probability this claim can be made. Besides testing the null hypothesis a number of other inferential statistics techniques exist to investigate rival hypotheses, the relationships between independent and dependent variables, systematic and non systematic variations, variance, and the effects of mistakenly accepting or rejecting an hypothesis.

2.1.5 Reporting of Results

A number of authors (McBurney, 1994; Ray, 1993) stress the importance of the reporting on results as part of the research process. The reporting on results can take various forms and is not limited to formal written reports like journal articles (McBurney, 1994; Stangor, 1998). Oral presentations or posters contribute to exchange of information in the informal network of researchers (Stangor, 1998). Visual illustrations in the form of tables or graphs can contribute greatly to convey findings (Tuckman, 1999; Agnew and Pyke, 1978). While the mentioned authors only refer to paper based reporting Sommer and Sommer (1991) suggest the use of video or audio recordings to present material to participants and to illustrate talks.

2.2 Qualitative Analysis Methods

The literature on qualitative research methods describes a wide variety of approaches. Henwood (1996) writes about the data display approach, content analysis, protocol analysis, grounded theory, ethogenics, discourse analysis, narrative analysis, and feminist research. Banister, Burman, Parker, Taylor, and Tindall (1994) identify the major approaches as observation, ethnography, interviewing, personal construct theory, discourse analysis, and action research. Tesch (1990) has identified over 40 types of qualitative research including for example action research, case study, content analysis, ethnography, and oral history. In the following sections the main characteristics common

to most qualitative research methods are highlighted. After a general description of the qualitative research process this literature review focuses on qualitative analysis with topics like data description, data analysis and interpretation. Validity and reliability in qualitative research are discussed, followed by a section on reporting of research results. A section on the use of computers concludes the review of the qualitative analysis methods.

In general terms qualitative research can be described as involving intensive work considering many aspects of data relating to a limited number of cases. It aims at discovering relationships and developing understanding. A number of authors provide general descriptions of qualitative research. Gay (1996) writes that qualitative research methods are applied to discover how people feel about things and what they believe. Qualitative research acknowledges that behaviour is significantly influenced by the environment in which it occurs. This leads to intensive data collection, stretching over long time and many variables, typically performed in a naturalistic setting concentrating on a single unit. The data are collected in the form of participant observation and in depth interviewing and are supplemented by additional relevant material. Qualitative analysis aims at a holistic, in-depth understanding of the phenomenon under study. Ragin (1994) emphasises the in-depth examination of few cases which considers many aspects and aims at the identification of key features. Ragin further writes that qualitative analysis often builds on the collected data and attempts to enhance these data. Merriam (1998) lists five characteristics which are common to all forms of qualitative research. Qualitative research attempts to discover the insider's, that is the participant's, perspective as compared to the outsider's, the researcher's, perspective. Data collection and analysis are primarily conducted by the researcher in person. Only in rare cases can a qualitative study be conducted purely based on documents. In most cases qualitative research involves some form of field work for data collection. Qualitative research is primarily an inductive process which builds hypotheses or theories from the data. Finally, the outcomes of qualitative research are presented in a rich, descriptive format containing words and pictures. Neuman (1994) writes that researchers applying qualitative analysis methods are trying to develop a deep and comprehensive understanding of the social world they are working in. In examining and organising observable data with qualitative analysis methods the researchers are trying to bring their ideas and theories into harmony with the

deeper structures below the observable surface. According to Silverman (1993) qualitative research is characterised by taking the subjects' perspective and trying to understand actions and meanings in their social context. The design of qualitative research is relatively unstructured and open. Qualitative research tries to avoid settling on concepts and theories at an early stage in the research process. Qualitative researchers prefer to work in everyday contexts to observe and examine the subjects in their natural environments. Closeness of the researcher to the data is seen as very important while quantification of data plays a subordinate role in qualitative analysis.

The process of qualitative research is characterised through intensive work on the data to develop ideas and concepts which can be further developed into theories. The research process is not linear, it is cycling through phases of data collection and analysis. There is usually no set hypothesis at the start of the study. Qualitative research allows space for emerging concepts. Krathwohl (1997, p.242) identifies the stages of the qualitative research process as gathering data, summarising and packaging data, finding gaps in data, repackaging and selecting central data, developing and testing propositions and constructing explanatory framework, and report writing. Krathwohl further divides the qualitative research activities into data, analysis and report streams and emphasises the interaction between these streams throughout the study process. The following more detailed description of the qualitative research process is based on Miles and Huberman (1994).

Development of Conceptual Framework

The qualitative research process starts with the development of a conceptual framework for the examination of the proposed research questions. The researcher designs plans for the sampling of data and the instrumentation. At all stages of the qualitative research process different components of the process mutually influence each other. The qualitative research process underlies no strict rule system, it always allows the integration of new thoughts and the modification of previously developed structures.

Data Collection

The next step in the process is the data collection, using methods like interviews or open ended questionnaires. Additional data collection can occur in later phases of the research process when for example evolving ideas suggest further analysis directions. At this stage

it is advisable to build a well-defined data management system which can be, with possible modifications, used throughout the research process. Besides the management of the data to be analysed, it is important to keep track of the data resulting from the analysis process, that is preliminary results and information about the steps taken to analyse the collected data.

Data Reduction in Preparation of Analysis

In preparation of the analysis, written transcript of audio recordings are produced to make the data more easily accessible. The extensive volume of the data is reduced by writing summaries of the collected data.

Application of an Initial Coding Scheme

The researcher starts to develop a coding scheme. The coding scheme is both influenced by the research questions and the initial information extracted from the data. The coding scheme is used to code the data to be able to extract and to link data segments. Again, the coding scheme is not a final structure, it develops and grows with the analysis process.

Drawing and Verifying of Conclusions

The analysis proceeds with the development of first descriptive and then explanatory displays. The researcher is drawing and verifying conclusions. The analysis is cycling through phases of specifying new conclusions, searching for proof within the data, revising conclusions, developing new ideas and formulating further conclusions.

Summarising of Results

Once sufficient analysis has been done on the data of one case, the researcher summarises the results and might proceed with the analysis of further cases, utilising the developed analysis structures. Based on a number of with-in case analysis results, the researcher can draw across-case conclusions to develop wider theories.

2.2.1 Qualitative Data

The data in qualitative analysis are usually very rich and complex. The data are describing people, objects and situations and are therefore of a very diverse nature. Qualitative data are mostly non-numeric (Huberman and Miles, 1994) and are expressed through words (Merriam, 1998). Qualitative data can consist of participants' accounts about their

feelings or experiences, of descriptions about participants' behaviours or activities provided by an observer, or of sections of collected documents (Patton, 1990). The sampling of data in a qualitative study mostly concentrates on a limited number of cases which are selected because they carry certain characteristics. The sampling is therefore not a random process but follows the purpose of the study in providing a characteristic or typical example (Gay, 1996).

The three main data collection methods used in qualitative research are participant observation, interviews, and the examination of existing documents (Wolcott, 1994; Merriam, 1998). Observation is a major form of data collection in qualitative research and provides the researcher with a unique insight in the phenomenon studied. It is often combined with the two other main forms of data collection. In participant observation the role of the researcher as participant and observer has to be carefully defined. The researcher has to be aware of the effects their presence and involvement can have on a situation or setting. The observations have to be recorded for later analysis. This can be as descriptions, direct quotations, or comments in forms of words or sketches. Observations can as well be recorded on video or audio tapes if technically possible and not too obtrusive (Merriam, 1998). Interviews in qualitative research can range from structured to unstructured formats. Most commonly used are semi-structured interviews for which a number of specific key questions and a range of more open-ended questions or topic areas are prepared. The role of the researcher is to elicit meaningful data by asking hypothetical, devil's advocate, ideal position, or interpretive questions. It is advisable to record the interview to preserve as much information as possible for the later analysis. The recording is commonly done on audio tape and in some cases, where visual information is of importance, on video tape (Merriam, 1998). Qualitative researchers frequently collect documents like minutes of meetings, reports, policy statements, or photographs to supplement data gained through direct observation. These documents contain additional information which might not be available through direct observation and has been recorded without being affected by the presence of the observer. The additional data collected can be of numerical form (Gay, 1996).

In many cases the audio or video data, resulting from the recording of observations and interviews, are not analysed directly. The researchers produce textual transcripts of the verbal data to be able to use available text-based analysis tools. The researchers have to

be aware that these textual data are derived from the original data and therefore might have been already influenced by some reduction in early stages of the analysis process (Huberman and Miles, 1994). Different transcription systems have been developed to capture, additionally to the sequence of spoken words, information like length of pauses, intonation, or emphasis. It is often helpful to work in parallel with transcript and original recording (Potter, 1996).

The amount of data to be analysed is usually very large in a qualitative study. Additionally, the researchers are adding further data during the analysis process to support the investigation of new questions arising from the analysis. Data collection and data analysis are tightly linked in qualitative research. To keep track of all data and processes involved it is essential to develop a comprehensive data management system. Ideally this system should be designed before the first data are collected and should be modified in later stages of the analysis process to meet all data management requirements. The task of the data management system is to facilitate storage of and access to the original data, to store the derived data documents so to be able to relate them back to the original documents, to keep track of the steps in the analysis process and to document preliminary and final analysis results. All data and results have to be accessible during the analysis process as well as after completion of the study for reference and verification (Huberman and Miles, 1994).

2.2.2 Analysis and Interpretation

In the description of the qualitative research process different authors put varying emphasis on the terms ‘analysis’ and ‘interpretation’. Wolcott (1994) writes about the process of ‘qualitative inquiry’ and labels elements of this process with the terms description, analysis, and interpretation. The descriptive element of the inquiry process attempts to describe what is happening based on the observational data, the analysis identifies essential features and relationships in these data explaining how or why things work while the interpretation addresses questions of meaning and context. Gay (1996) uses the terms analysis and interpretation. Gay outlines analysis as a process of looking for categories, patterns, and themes in the data with the objective of facilitating a coherent synthesis of the data and interpretation as a process of drawing conclusions or gaining insights. Interpretation is described as a continuous effort which takes places throughout

the duration of the study. Merriam (1998) only uses the term analysis which she describes as a process of consolidating, reducing, and interpreting the data with the aim of creating meaning or making sense. Merriam distinguishes between three levels of analysis which range from concrete description to high-level abstraction and calls these levels descriptive accounts, category construction, and theory construction. Huberman and Miles (1994) describe two different levels of understanding in qualitative analysis. One level is the descriptive aspect where the researcher is exploring what has happened and how it has happened. The other level is the explanatory aspect where the researcher is looking at the question of why something has happened. In the following paragraphs of this section the term ‘analysis’ will be used to refer to the whole qualitative analysis process comprising the elements of description, analysis, and interpretation.

As indicated in the section on the qualitative research process (see introduction to Section 2.2) qualitative analysis is closely linked to data collection. Data collection and analysis are not two separate phases in the research process but strongly influence each other (Huberman and Miles, 1994; Merriam, 1998, Tindall, 1994). As the analysis progresses the familiarity of the researcher with the data increases so new questions arise and produce new ideas for investigation. To follow up these new analysis aspects, the researcher often has to collect further data (Huberman and Miles, 1994).

Neuman (1994) describes six steps which are common to most forms of qualitative data analysis. The first step is the detailed familiarisation with data and notes. This is followed by the grouping of details into organising ideas. Then the researcher looks for new ideas, based on the notes and the grouping from the previous step. In the next step the researcher is trying to put the formed ideas into relationships to each other, grouping together ideas with logical similarity. Then larger groups are formed by comparing and contrasting sets of ideas. Lastly these groups are reorganised and linked together according to broader integrating themes. In summary the process is one of building up from specific details to an overall set of logical relationships. Merriam (1998, p. 178) describes qualitative analysis as “a complex process that involves moving back and forth between concrete bits of data and abstract concepts, between inductive and deductive reasoning, between description and interpretation.” Qualitative analysis involves the construction of categories based on the constant review of data within the theoretical framework of the study. Categories are abstractions derived from the data. The development of theories

involves the integration and refinement of categories, properties, and hypotheses. This process includes both induction and deduction, as theories are developed and tested against the data (Merriam, 1998). The development of theories is an abstraction process, where the researcher moves from the observable data to the unobservable theories (Miles and Huberman, 1994).

To perform the qualitative analysis process the researchers apply a range of techniques like data reduction, system closure, coding, memo writing, and display of data. Qualitative studies are characterised through the large amount of data collected. The data are rich and often unstructured. To be able to reveal and concentrate on the most important aspects in the data, researchers combine and link data, remove notes and details. This technique is called data reduction (Krathwohl, 1997; Huberman and Miles, 1994). System closure refers to the technique in qualitative analysis of the feedback of intermediate analysis results into the analysis process for investigation alongside the original data (Huberman and Miles, 1994; Richards and Richards, 1994).

Coding is a process of critically analysing the data, identifying themes and topics which represent categories. Codes represent both the description of observed events and the interpretations or comments of the researcher (Gay, 1996). Coding is both a mechanical task and a creative process which occurs on two levels. On the first level it is a data management tool which helps to identify interesting data. On the second level it is an analysis tool which assists in the capture of thought and ideas (Merriam, 1998). During the analysis process the coding structures have to be refined, extended and related to each other as additional material is explored (Pidgeon and Henwood, 1996). Neuman (1994) defines three kinds of coding. The open coding is the first pass through the data. With open coding the researcher attempts to develop preliminary concepts. The researcher is open to new ideas. After the open coding the researcher compiles a list of themes. The second pass through the data is called the axial coding. The starting point for the axial coding is an organised set of initial coding structures representing preliminary concepts resulting from the open coding. The purpose of the axial coding is to review and examine the initial codes and to organise ideas or themes. The researcher starts to ask about cause and consequence, conditions, interactions, strategies and processes. Based on the further work with the data, initial operational definitions or concepts are combined or separated. In the third phase of coding, the selective coding, major themes of the research project are

identified. The concepts are well defined and the researcher is starting to organise the overall analysis around several core generalisations or ideas. The researcher is looking selectively in data and codes for cases that illustrate concepts and/or provide comparisons or contrasts.

Another analysis technique, applied in parallel to the coding, is the technique of writing memos. A memo can contain any kind of information like explanations of categories, comments, references to examples, links to literature, or emerging theories. It is important to capture ideas immediately as they arise to prevent them from being forgotten later on (Pidgeon and Henwood, 1996). The capture of intermediate results in the form of memos supports the transition from data to theory and therefore contributes to theory building (Huberman and Miles, 1994).

A further important technique in the analysis process is the display of the data. Graphical presentation can show the data in a light which emphasises trends and relationships or indicates which data do not follow a general pattern (Krathwohl, 1997). A full picture of the available data can reflect evolving concepts and theories. The data display and the analysis process influence each other (Huberman and Miles, 1994). Data can be presented in a variety of forms like spatial or temporal maps, typologies, sociograms, flow charts, organisational charts, causal diagrams, lists and grids (Neuman, 1994).

2.2.3 Validity and Reliability

The terms of validity and reliability have their origin in quantitative research yet have to be seen in a different light in qualitative research (Merriam, 1998; Burns, 1997). Qualitative research is characterised by a small number of cases studied intensively and a close involvement of the researcher throughout all phases of the research process from data collection to analysis and interpretation. Validity and reliability in qualitative research can therefore not be achieved through approaches like large sample numbers or independent observations. Qualitative research has to build on a careful conceptualisation of a study including all steps of data collection, analysis, interpretation, and presentation of the findings (Merriam, 1998). Smith (1996) goes further in saying that qualitative research should develop its own criteria for judgement instead of taking over the quantitative approach. The review presented in this section follows the approach of discussing the terms of *reliability*, *internal validity*, and *external validity* in context of the

qualitative research process. Following that a number of techniques to increase validity and reliability in qualitative research are discussed.

A study is usually considered as *reliable* when its findings can be replicated. Replication in a qualitative context is problematic because human behaviour is never static and studies are conducted in a social and interpersonal context. The researcher does not attempt to analyse isolated relationships between single variables but attempts to describe and explain the experiences of individuals (Burns, 1997; Merriam, 1998). As *reliability* cannot be achieved through replication it has to be enhanced for example through triangulation, the employment of well-trained observers, an investigator position explaining assumptions and theories behind the study, and an audit trail carefully outlining all steps of a study (Merriam, 1998). The reliability of a study can also be positively affected by the recording of behaviour on video or audio tape because recorded behaviour can be observed repeatedly by multiple researchers (Peräkylä, 1997). The *internal validity* of a study is seen as the match between the research findings and the reality. To establish this match is difficult in qualitative research because the qualitative concept allows for many forms of reality, acknowledging the possibility of different perspectives of reality for different people (Merriam, 1998). In qualitative research the researcher is very close to the data, participants, or situations. While this conflicts with the quantitative viewpoint of ‘independent observation’ the closeness of the researcher can be seen as strength (Burns, 1997; Merriam, 1998). *External validity* in a quantitative context refers to the generalisability of the research findings. Generalisability is not the focus of qualitative research which deliberately concentrates on the intensive study of a limited number of cases. External validity in a qualitative sense can be facilitated through two approaches. Firstly, the researcher can look at quantitative aspects within one case, by for example studying a number of subjects within the case. Secondly, the researcher can consider theories resulting from a study as working hypotheses, valid within the local context and starting point for work in a different context. External validity in this qualitative sense can be supported by measures like providing rich, thick descriptions of a study to fully inform others, by emphasising the typical characteristics of a study to facilitate comparison with other studies, or by utilising multiple site designs to maximise diversity (Merriam, 1998).

Suggestions to increase the validity and reliability in qualitative research are linked to *internal coherence, presentation of evidence, independent audit, triangulation, member validation, and reflexivity* (Smith, 1996):

- To achieve internal coherence the researcher conducting a study constantly has to evaluate if the study is conducted in a consistent and coherent manner. It has to be tested if the data warrant the interpretations or if alternative interpretations have to be considered (Smith, 1996).
- The presentation of evidence has to contain enough raw data to allow the reader to see how the interpretations are grounded in the data. A research report should allow the reader to distinguish between raw data and interpretation (Smith, 1996).
- An independent audit means that a researcher, originally not involved in the study, will attempt to follow the argumentation and interpretation in a study based on the material collected and the direction taken. This researcher will not be looking for alternative accounts but will test the validity of the particular account given (Merriam, 1998; Smith, 1996).
- Triangulation stands for employing a combination of methods, investigations, and perspectives to facilitate richer and potentially more valid interpretations (Tindall, 1994; Burns, 1996; Merriam, 1998; Smith, 1996).
- Member validation means that analysis results are presented to the participants of a study for comment and verification. This can be done at various stages during the research process (Smith, 1996; Merriam, 1998).
- Reflexivity refers to the challenge presented to the researchers to constantly monitor their own involvement in a study. The researchers have to reflect on their own sets of values, on the relationship of their study to other work, or on the influence their interference has on the study (Tindall, 1994; Smith, 1996).

2.2.4 Reporting of Results

The main purpose of a research report is to share the knowledge gained in a study with others. A secondary purpose is that writing the report helps to clarify thoughts and arguments (Burns, 1996). Burns (1996, p. 397) describes the research report as “a logical, descriptive and analytical presentation of evidence that has been systematically collected and interpreted.” Qualitative research does not prescribe the format of the research report,

a number of different forms, dependent on target audience or study method, are possible (Merriam, 1998; Banister, 1994; Burns, 1996).

One of the challenges of writing a qualitative research report is to decide how much material to include in the report. Evidence which can not be included in the report should be retained for reference by the researcher who conducted the study or by other researchers. This material might become a valuable database for later research (Burns, 1996). While a research report is usually of written form it can as well be of oral or pictorial form (Merriam, 1998). Raw material can be included in an appendix (Banister, 1994). The display of data in chart, matrix, table, or figure format can help the reader to more quickly understand the conclusions drawn by the researcher (Merriam, 1998). The report can be given a sense of reality and credibility by the inclusion of pictures, video or audio recordings, artifacts, or documents (Burns, 1996).

2.2.5 Use of Computers

The use of computers to assist qualitative analysis has become widely accepted (Kelle, 1995; Gay, 1996; Krathwohl, 1997; Smith, 1996; Durkin, 1997; Lee and Fielding, 1991; Tesch, 1991; Richards and Richards, 1994). Despite the diversity of the different methodologies and approaches in qualitative research all qualitative analysis combines the two parallel elements of conceptual and mechanical tasks. Computer systems provide invaluable support for the mechanical tasks in the analysis process (Tesch, 1991). Using computer systems for the tedious, mechanical tasks leaves the researcher with more time for the creative aspects of the analysis process (Durkin, 1997). To assist with the mechanical tasks is the intention behind the development of qualitative data analysis computer systems. The researcher applying these systems has to be aware that the systems only support but do not perform the analysis. The responsibility for study design and conceptual work still lies with the researcher (Smith, 1996; Durkin, 1997; Krathwohl, 1997).

Computer systems provide support for two central tasks of the analysis process, the data management task and the task of coding and retrieval. Qualitative analysis deals with large amounts of raw data and data produced as part of the analysis process. Computer systems are ideally suited to assist in the management of these data (Krathwohl, 1997; Akeroyd, 1991; Durkin, 1997; Richardson and Richardson, 1994). The task of coding and

retrieval is facilitated by computer systems in several ways. Beside providing some automated features for searching for words or phrases the main advantages of using computer systems as compared to a pen, paper, and scissors approach are the flexibility of adding or changing codes, the possibilities of combining codes into categories, of coding data in several ways, or of graphically presenting coding structures (Durkin, 1997; Krathwohl, 1997; Tesch, 1991). A number of authors argue that using computer systems to assist the coding allows the researcher to remain closer to the raw data, because instantaneous swapping between codes and data is possible, and coded data can always be seen in context (Lee and Fielding, 1991; Durkin, 1997). The qualitative analysis process, in which data collecting and coding overlap, is well supported by the interactive nature of computer systems. Links between data and emerging ideas can be explored (Durkin, 1997; Richardson and Richardson, 1994). The use of qualitative analysis systems allows one to analyse data more quickly and thoroughly (Durkin, 1997). Krathwohl (1997) and Richards and Richards (1991) write that the use of computer systems in qualitative analysis encourages the application of quantitative techniques on qualitative data which can enrich the qualitative analysis process.

The advantages of using computer systems for data management and coding are well established and there is potential to assist qualitative analysis in further areas. So far qualitative analysis systems mainly concentrate on the analysis of textual data. Support for analysing video, audio, and image data would be of benefit to qualitative analysis researchers (Weaver and Atkinson, 1994; Krathwohl, 1997). The facility of storing data on a computer can enhance research performed in teams. Data can be accessed and annotated by multiple researchers working within a team (Durkin, 1997; Lee and Fielding, 1991). This is linked to the possibility of creating data archives of modified material which have access restrictions in their original form. Data identifying individuals, for example, can be made anonymous so that the major part of the material can be released for open access in the research community (Fisher, 1994). Lee and Fielding (1991) see a potential use of computers in creating simulated social worlds for the teaching of sociology.

2.3 Comparison of Quantitative and Qualitative Analysis Methods

Quantitative and qualitative analysis methods follow relatively different approaches to analysis. They emphasise different aspects of the analysis process and aim at different analysis results. Based on the different strengths and weaknesses of the methods, a number of authors argue for the combination of both methods (Howe, 1988; Brannen, 1992; Hammersley, 1992; Bryman, 1988; Bryman, 1992; Greene, Caracelli and Graham, 1989; Patton, 1990; Miles and Huberman, 1994; Reichardt and Rallis, 1994; Dey, 1993). Howe (1988) stresses that quantitative and qualitative analysis methods are different but not incompatible.

In the context of combining methods it is important to distinguish between data and methods. Quantitative and qualitative data are frequently used within the same study (Howe, 1988; Hammersley, 1992) and in many studies qualitative aspects are quantified and quantitative claims are expressed in qualitative, verbal form (Hammersley, 1996). Bryman (1992) argues that the combination of both methods, looking at issues like intention of the study, data collection methods, analysis design and construction of results, leads to a more comprehensive approach.

In the next section quantitative and qualitative analysis methods are compared concentrating on their differences. This is followed by a section highlighting the common elements of the methods and suggesting how both methods can be combined in a study.

2.3.1 Differences between Quantitative and Qualitative Analysis Methods

The comparison between quantitative and qualitative analysis methods presented in this section follows the outline of Burns (1997) who contrasts the methods under the headings of assumptions, purpose, method and role of researcher.

Assumptions

Quantitative analysis takes a positivist perspective. It follows the assumption that sociological concepts can be conceptualised as variables and that important features of the social world can be captured with the help of objective, precise measures. Qualitative analysis follows largely a non-positivist perspective. It focuses on subjective meanings, on definitions, metaphors and symbols and on descriptions of specific cases (Neuman, 1994). The presumption in quantitative analysis is that variables can be identified and

measured whereas in qualitative analysis variables are seen as complex, interwoven, and difficult to measure (Burns, 1997). The quantitative approach to enquiry is deductive and outcome-oriented while the qualitative approach is inductive and process-oriented (Gay, 1996). The researchers following the quantitative approach build their analysis on the assumptions that the research topic can be captured with the set of variables identified and the measurement methods developed. With the qualitative approach researchers attempt to keep the investigation open and to be sensitive to unanticipated features of the study, which might arise out of the context or of the descriptions and explanations of the subjects involved (Howe, 1988).

Purpose

Burns (1997) describes the purpose of quantitative method as prediction, generalisation, and causal explanation. As contrasting terms for qualitative method he lists interpretation, contextualisation, and understanding the perspectives of others. In describing their overall purpose Gay (1996) writes that both methods aim to explain phenomena. The difference comes where quantitative research tries to predict and/or control phenomena through focused collection of numerical data where qualitative research attempts to gain insight in and understanding of phenomena through intensive collection of narrative data. Brannen (1992) contrasts the aim of generalisation in quantitative research with the focus in qualitative research on creating studies which allow the replication of findings in other similar cases or sets of conditions. Denzin and Lincoln (1998) write that quantitative research builds abstractions of the social world by deriving data from the study of large numbers of randomly selected cases. They contrast this with the view of qualitative research which collects rich descriptions of the social world and is deeply concerned about the individual's point of view.

Method

The role of hypothesis and theory differ among both research methods. While quantitative research commences with fully developed hypothesis and theory, qualitative research characteristically concludes with a hypothesis and theory grounded in the phenomena studied (Burns, 1997). A hypothesis in quantitative research is specific, testable, and stated prior to a particular study whereas a hypothesis in qualitative research is tentative, evolving, and based on a particular study (Gay, 1996). The quantitative research process

is clearly designed before the study is started. The researcher has stated precise research questions, has scheduled sharply distinguished data collection, analysis and write-up phases and has selected the statistical analysis procedures to be employed. Qualitative research starts with a far less specified schedule. The research question is formulated in a provisional form. The researcher has defined the parameters for the initial data collection and has scheduled time for the analysis phases (Howe, 1988). Quantitative research follows a linear path with clearly separated phases of the process. The analysis starts only once all data are collected and condensed into numbers. The qualitative research process is cyclic. It is directed towards construction of meaning. The phases in the research process are not clearly separated, the analysis starts soon after the first data are collected and continues throughout most of the research process. Analysis and data collection constantly influence each other (Neuman, 1994).

In quantitative studies data collection methods like structured interviewing, self-administered questionnaires, experiments, structured observation, content analysis and the analysis of official statistics are used. Qualitative studies work with participant observation, largely semi- and unstructured interviewing, qualitative examination of texts and language based techniques like conversation and discourse analysis (Bryman, 1992). The raw data collected are largely numbers in quantitative research and words in qualitative research (Gay, 1996; Neuman, 1994). The assumption in quantitative analysis is that aspects of social life can be measured using numbers. The researcher applies statistical methods to manipulate numbers that represent empirical facts in order to test an abstract hypothesis. Qualitative analysis is less abstract than statistical analysis, the researcher is much closer to the data which are verbal or textual descriptions of social life. Qualitative research is often inductive without a clearly specified outcome in the beginning (Neuman, 1994). The quantitative researcher analyses data in the view of extracting values for the variables identified in the analysis design. In the further analysis process the researcher concentrates on these variables and uses them for the testing of the study hypotheses. For the qualitative researcher the data are the centre of attention. While the researcher uses tools like coding to assist conceptualisation or data reduction the focus will remain on the original data (Brannen, 1992). Based on the analysis with statistical methods the reporting in quantitative research occurs in the language of statistics. In qualitative research, where the analysis is based on textual data, the reporting of results is

done in form of a descriptive write-up, using the language of the participants of a study (Burns, 1997).

Using quantitative methods it is possible to measure responses of many people to a limited set of questions, that is to gather specific information on many cases. This approach facilitates the comparison and statistical aggregation of the data which results in a broad, generalisable set of findings. Qualitative research captures a wealth of detailed information about a much smaller number of people and cases. It promotes the understanding of cases and situations for the price of reduced generalisability (Patton, 1990). Quantitative analysis uses statistical techniques like probability theory or statistical correlation to test the generalisability of findings. To allow generalisation samples have to be selected randomly out of a population and the number of samples needed to reflect the population is determined with methods based on statistical inference. Qualitative analysis does not aim at generalisability based on statistical calculations. It is concerned with the possibility of replicating findings in other similar cases or under a similar set of conditions. The samples in qualitative studies are selected using theoretical criteria. There is no strict guideline about how many cases have to be examined. The researcher has to balance between theoretical saturation and available time and money (Brannen, 1992). In quantitative studies the validity depends on careful instrument construction and the appropriate use of the instrument. The researcher has to make sure that the instrument measures what it is supposed to measure. In qualitative studies the researcher is the central instrument. The validity of the findings is highly dependent on skill, competence and rigour of the person doing the field work (Patton, 1990).

Role of Researcher

The role of the researcher in both types of analysis has a different emphasis. In quantitative analysis the researcher is detached from the phenomena studied. The researcher applies formal instruments and aims for objectivity throughout the study. In qualitative analysis the researcher is closely involved throughout the phases of data collection and interpretation. The researcher tries to see events from the informant's perspective and develops an emphatic understanding for the phenomena studied. The researchers themselves can be seen as instruments Burns (1997). Taking on the role of an instrument, the qualitative researchers must not only attend to the data but as well to their own cultural assumptions (Brannen, 1992). The relationship between researcher and

subject is distant in quantitative research and close in qualitative research. The quantitative researcher takes an outsider position in relation to a subject whereas the qualitative researcher takes the role of an insider (Bryman, 1988).

2.3.2 Commonalities of Quantitative and Qualitative Analysis Methods

The literature shows different approaches and arguments for combining quantitative and qualitative analysis methods. Hammersley (1992) argues that the two methods are not as different as they are commonly seen and says that discussions should not concentrate on an either-or decision between quantitative and qualitative. Howe (1988) writes that the two methods are so closely linked that each study has both elements and only the emphasis on one or the other method differs from study to study. Neuman (1994, pp. 316) says that “Social researchers systematically collect and analyze empirical evidence in order to understand and explain social life” and than goes on to discuss the different emphasis set by quantitative and qualitative researchers. Brannen (1992), Bryman (1992) and Greene, Caracelli, and Graham (1989) discuss the combination of both methods, either in an integrative approach of both methods to develop the findings of a study from different viewpoints or in a complementary approach of both methods to use the strength of each method to look at different aspects of a study. Concluding her paper on the combination of both methods Brannen (1992, p. 32) writes about “the small amount of literature on combining methods and the relatively few studies which have adopted this strategy” and emphasises the theoretical and practical potential of combining both methods.

Common Elements in Both Methods

A number of arguments are commonly used to prove the distinction between quantitative and qualitative analysis methods, but it can be argued that the methods are not as different as it might seem. Hammersley (1992) discusses a number of points to show that, at closer look, both methods share common ground. One issue is the dominant use of quantitative or qualitative data, that is numbers or words, in the respective methods. Despite this generally valid distinction qualitative researchers regularly make quantitative claims expressing measurements in verbal form using expressions like regularly, frequently, often or sometimes. Quantitative researchers on the other hand quantify qualitative data by assigning numerical values to qualitative concepts. A further argument used for the

distinction is that quantitative studies conduct experiments in artificial settings whereas qualitative studies are carried out in natural settings which provide the only way to observe natural behaviour. However, this distinction can be misleading. One can ask how natural settings like classrooms or court rooms are and how strong the influence of a researcher is in such a setting. Quantitative research is commonly associated with hypothesis testing whereas qualitative research is usually seen as hypothesis building. Despite this general association all research involves both induction and deduction in the broad sense of those terms. All research moves from data to ideas and from ideas to data.

Weaver and Atkinson (1994) say that the primary purpose of coding in quantitative and qualitative research is the same. Coding is used to structure and facilitate analysis. It is used to segment and describe data for later retrieval of related segments.

Howe (1988) writes that there is no absolute distinction between quantitative and qualitative interpretation of results. Studies invariably mix the different types of interpretation. What varies is the emphasis on one or the other type. Quantitative researchers are restricted by earlier decisions about what is measured and how it is measured. Qualitative researchers are looking for new relationships, new ways of aggregating and coding data not envisioned in the original design. Beside that, both researchers are constructing arguments based on evidence. The statistical analyses conducted by the quantitative researchers are merely mechanical inferences in a much larger set of knowledge claims, assumptions and instances of nonmechanical inferences. At each level of the research process, looking at data, design, analysis and interpretation of results, the two methods are inextricably connected. Each study has both quantitative and qualitative elements, the difference is in the emphasis on one method or the other.

Neuman (1994) shows that both methods, while handling them differently, deal with similar issues. He states that the topic of researcher integrity is of importance in both methods. Researchers of both types of methods have to build up the trust of the readers. Quantitative researchers build on standard techniques and statistics to demonstrate the integrity of their research, qualitative researchers very carefully reflect on and check the evidence and present their finding with sufficient detail for the reader to follow their argumentation. Both, quantitative and qualitative analysis, deal with the concept of triangulation. Quantitative analysis uses triangulation to increase the objectivity in testing

hypotheses and to reduce method effects. Qualitative analysis applies triangulation to disclose richness and diversity. Like other authors, Neuman writes about the value of using qualitative data in quantitative research and vice versa.

Combination of Both Methods

According to Greene, Caracelli, and Graham (1989) the combination of quantitative and qualitative analysis methods has two main aspects. One is the aspect of triangulation as the attempt to investigate a research question from different perspectives with the intention of arriving at convergent results. The other aspect is multiplism being the intention to use multiple methods to look at different aspects of a study and to produce complementary results. The significance of triangulation is the use of multiple methods to enhance the validity of findings and the degree of confidence in the findings (Bryman, 1992). This is based on the assumption that all methods have inherent biases and limitations which will be reflected in the results of a study. Using more than one method within one study will lead to a richer analysis result which will be less influenced by the specific characteristics of one method (Greene, Caracelli, and Graham, 1989). The benefits of triangulation achieved by combining methods are, as well, supported by Pyke and Agnew (1991). A combination of methods in the sense of multiplism assists in looking at macro and micro levels of a study. Quantitative methods often investigate large scale structural features of social life, while qualitative methods tend to address small-scale behavioural aspects (Bryman, 1992).

There are several ways of combining the two methods in a research process. The research process can be dominated by the quantitative method with the qualitative method in a supporting role, or it can be dominated by the qualitative method with the quantitative method supporting or both methods can have equal weight. Bryman (1992) discusses these options:

- In the first case, where the qualitative methods play a subordinate part in the research process, the qualitative work is typically carried out prior to the main quantitative study. The qualitative method can be used to generate data through in-depth interviewing. These data, produced with a qualitative method, are then quantified and explored using quantitative analysis methods. The data are not explored in a qualitative sense. Further, qualitative methods can be used in the development and piloting of quantitative research instruments like questionnaires, scales and indices. In

case the dominating quantitative part of the study is carried out first, a qualitative method can be used for interpretation and clarification of quantitative data. The researcher can use the qualitative method to follow up on an issue where quantitative methods are not appropriate, to look at a small but interesting subgroup in more detailed exploration or to clarify a puzzling finding.

- If the qualitative method is dominant, the quantitative part of the study can provide the quantitative background data in which to contextualise small scale intensive studies. The quantitative work may provide the basis for the sampling of cases and the comparison groups for an intensive study. Quantitative methods may, as well, be used to test hypotheses built by qualitative work.
- Where both methods get equal emphasis in the research process two separate but linked studies, which are distinct from each other throughout the research process, can be carried out. The results of both studies can be combined to provide a general picture. Alternatively, the two methods can be integrated into one study. The link between the methods can occur in the data collection, analysis or writing-up phases. The methods can be conducted simultaneously or consecutively.

The arguments for combining quantitative and qualitative methods brought forward so far looked at the research process as one unit. Hammersley (1996) takes a slightly different approach. He distinguishes five aspects of the research process: "formulating the problem, selecting the cases, producing the data, analysing the data and communicating the findings" (Hammersley, 1996, p. 173) and states that these aspects apply to all forms of research. From this starting point he argues that one should look at each stage of the research process separately, testing which method or combination of methods would be most suitable for the particular stage.

The increasing use of computers in qualitative research might facilitate the integration of quantitative and qualitative data and research methods. The use of computers makes it possible to manage large amounts of rich data and to increasingly analyse qualitative data with automated procedures (Bryman, 1992; Dey, 1993).

2.4 Summary

In this chapter quantitative and qualitative analysis methods have been described and compared. It has been shown that both methods, while displaying differences in direction

and emphasis, deal with the same set of analysis steps. Table 2.1 summarises commonalities and differences, directing attention to issues important for the design of a multimedia information system to support the study of behaviour.

The following chapter will examine existing computer systems to support the study of behaviour coming from a wide range of disciplines and following quantitative or qualitative analysis directions.

	Quantitative Analysis	Qualitative Analysis
Data Collection	questionnaires, structured interviews, tests, non participant observations multiple samples multiple participants	semi-structured and unstructured interviews, participant observations, additional material comparatively few samples comparatively few participants
⇒ <i>Multimedia Behaviour Recordings</i>	material collected in text, data, video and audio formats	
Description	measurement of variables predefined few variables many incarnations	open description flexible coding scheme evolving coding
⇒ <i>Description</i>	textual descriptions or codes describing aspects of behaviour descriptions linked to behaviour recordings	
Analysis & Interpretation	working with numbers descriptive and inferential statistics	reviewing, grouping of descriptions viewing of behaviour recordings
⇒ <i>Retrieval Interpretation</i>	retrieval of thematic groups of descriptions interpretation based on these thematic groups	
Reporting	reporting in language of statistics removed from individual data	reporting in language of participants rich details
⇒ <i>Reporting</i>	formulating findings supported by data from analysis and interpretation steps	

Table 2.1: Comparison of quantitative and qualitative analysis

Chapter 3

Existing Computer Systems to Support the Study of Behaviour

Computers are regarded as valuable tools to support the study of behaviour, conducted both with quantitative and qualitative research methods. Computers are used in quantitative studies of behaviour for event recording and data analysis. In event recording, computerised check sheets or more general interfaces to record behavioural events online increase the capacity and precision of observation input compared to paper based tools. Additionally, the data are immediately available for analysis and display, making the time consuming and error prone process of conversion from paper to computer unnecessary (Martin and Bateson, 1993). With additional hardware automatic recording of certain behavioural events is possible (McKenzie and Carlson, 1989).

Computers have become an essential tool for behaviour analysis. Computer systems can carry out sophisticated analytic procedures on large amount of data quickly and efficiently. The systems offer many possibilities for data presentation in table or graphical form (Hartmann and Wood, 1990). The use of statistical packages speeds up calculations, increasing accuracy and sophistication as compared to the manual processing of data (Bryman and Cramer, 1994).

Computers are regarded as necessary tools in qualitative research (Lee and Fielding, 1991; Tesch, 1990; Richards and Richards, 1994; Seidel, 1991). Computers are widely accepted for their data management capabilities (Tesch, 1990) and are influencing the thinking about qualitative methods (Richards and Richards, 1994). Computer systems provide valuable assistance in the data management tasks. The systems support working on text by locating words or phrases, searching in text, counting words or phrases, adding reference information, sorting text segments by topic or inserting key words or comments (Tesch, 1990)

The use of computers offers extended possibilities for analysis and alters the experience of research (Richards and Richards, 1994). Because software is assisting in code-and-retrieve and in re-coding after changes to the coding categories, more coding and re-coding can be done. The use of software leads to greater flexibility. Retrieval becomes more a theorising task because codes, description of codes, and coded text can all be looked at simultaneously. The building of a coding structure is part of the research process. The coding becomes a process which accompanies the whole study. Coding can be done on a general level first and can be refined later. The researcher can follow different approaches to coding by coding with multiple codes.

The following discussion focuses on computer systems which facilitate the analysis of recorded behaviour. Other types of computer systems like statistical packages, event recording systems or qualitative analysis systems (not related to the analysis of recorded behaviour as pursued in this research) have been omitted from the discussions. This distinction was made to be able to focus on system features essential for the development of a new concept for a *multimedia* information system to support the study of behaviour. It is acknowledged that a number of features of systems not described here would be very well suited to complement such an information system.

3.1 Existing Systems

Following are descriptions of a number of systems which support the analysis of behaviour captured in multiple data formats. Here only the main features of these systems are described. Later sections refer to single features of these systems in more detail and in the context of looking at a certain characteristic across a collection of systems. Further, the emphasis in this description is on features relevant to the development of an argument for a new concept for a multimedia information system to support the study of behaviour. Broad issues concerning the implementation of a system, like user friendliness, help system, implementation platform or performance, were put aside for the purpose of this work. Publications relating to the concepts of the systems were located or, where this was not possible, the concepts behind the systems were deduced from the system descriptions.

Each of the systems mentioned here has unique features or is an important representative for a system coming from a certain tradition. The OBSERVER is probably the most known and most used system originating from the area of quantitative behaviour analysis.

It supports digital video. CABER is interesting with its unique design of a formalised coding language for closed behavioural domains. MacShapa represents the area of cognitive studies and is of special interest because a number of research papers have been written which explain the concepts behind the system. The system vPrism was developed for studies in the education area. It supports the analysis of digital video recordings and their transcripts. Constellations is one of few qualitative analysis systems designed to analyse multimedia data. NUDIST is one of the most prominent representatives for a qualitative analysis system in the area of sociology. Designed for the analysis of textual transcripts it can be used together with CVideo to facilitate the analysis of video data. The qualitative analysis system ATLAS.ti was also originally designed for text analysis but has in recent versions been extended to handle video, audio, and graphics data. Systems like NUDIST and ATLAS.ti are important for the study of analysis techniques applied to text documents with the view of transferring these techniques onto the analysis of video data. The system Code-A-Text is a representative for a system which clearly focuses on the analysis of text but has been extended into a multimedia system. It has text analysis features which could possibly be transferred into a multimedia system in the form of speech or image recognition. FERAL supports the analysis of video recordings linked with expert commentary. It is a purely qualitative system which aims at analysis in an open domain.

3.1.1 OBSERVER

The Observer Video-Pro (further referred to as ‘OBSERVER’) consists of the Observer base package and the support package for video analysis. It is used to record and analyse events in human and animal behaviour. Information about the OBSERVER can be found in documents produced by the developer, Noldus Information Technology (1998a and 1998b), and in (Noldus, 1991).

OBSERVER is a system which started as an event recorder. Events can be recorded in live observations or from video. The video can either be played on a VCR which is monitored by the PC or the video can be played as digital video on the PC giving the user the advantage of random access to any position in the video file. The range of functions available in the OBSERVER system clearly show the system’s origin in behaviour analysis and modification. To prepare the coding the user chooses sampling and timing

methods and defines the independent variables. OBSERVER defines a strict structure for the coding sentences. This structure is ‘actor - behaviour - modifier1 - modifier2’. The ‘actor’ is the subject who is displaying the behaviour. In single subject observations, naming the actor can be omitted. The ‘behaviour’ describes the basic behavioural element to be observed. Behavioural elements are defined as events, without duration, or states, with measurable duration. A behavioural element can have up to two ‘modifiers’ assigned. Modifiers allow a more detailed description of a behaviour. They can specify a person as receiver of a behaviour, an object involved in the behaviour, the direction or intensity of a behaviour or other information. Through the design of its coding system with checks and feedback during the coding process, OBSERVER attempts to ensure the correctness of collected coding instances according to the defined coding parameters. The code instances are stored in an OBSERVER specific format in a text file. Code instances from several coding passes can be stored together. OBSERVER provides a number of different analysis procedures. The user can specify which observational data and time intervals are to be included in the analysis. ‘Actors’, ‘behaviours’ and ‘modifiers’ are specified for the selection of code instances. Wildcards and Boolean conditions can be used. Lag sequential analysis explores the temporal relationships between preceding and following events. OBSERVER calculates a range of elementary statistics like duration, frequency and latency for the selected coding instances. The statistical data can be exported in different formats. A time-event table produces a sequential, chronological listing of all recorded events. A time-event plot gives a graphical presentation of recorded events in relationship to elapsed time. Pairs of observational data files can be compared to calculate inter- and intra-observer reliability. Beside the export of statistical data OBSERVER can print event summaries, capture video images to disk and can compile a video play list of selected video segments. These data can be used in other systems to create reports on the analysis results produced using OBSERVER. Reviews of the OBSERVER can be found in Keene (1994) and Lazarus (1994).

3.1.2 CABER

The CABER (Patrick, 1985a and 1985b; Patrick, Ho, and John, 1985; Patrick and Chong, 1987) system was developed with two main analysis goals in mind. Firstly, to create a system which allows the description of the complex interaction of human behaviour in a way which makes these behaviour descriptions accessible to the analysis through a

computer system. Secondly, to provide a system which supports the capture of behavioural events in real time.

The first goal was achieved by the use of formal languages. Before behavioural descriptions can be captured using CABER a formal language has to be defined for the specific domain of the behaviour to be captured. This language attempts to provide descriptions for all possible behaviours within the domain in form of a formally defined LL2 grammar. The user has to produce the code instances in one coding pass. Only code instances which are syntactically correct sentences according to the grammar are stored as code instances. The semantic correctness of coding sentences is facilitated by and dependent on the careful specification of the formal language for the domain. The code instances are stored in text files in a CABER specific format, separately for each coding pass. The CABER query mechanism allows querying for behavioural descriptions as they can be built by definition through the description language connected by Boolean and sequence operators. The queries are applied to the data resulting from one coding pass. The query results can be used to calculate statistics and to view the according video sequences.

The second goal of capturing behaviour description in real time is supported by the use of a key tablet as data input device. The keys of the key tablet are overlaid with the words of the formal description language. The words are arranged according to their use in the sentences to facilitate fast data entry. The behaviour descriptions can be recorded based on live observation or on watching a video tape. In the case of live observation the behaviour descriptions are stored with a system time. Later, in retrieving code instances, these system times can be linked to video recording times if the behaviour observed has been video taped at the same time. This feature allows live observation combined with the possibility of reviewing selected behaviour sequences on video at a later point of time. The video is played as analog video on a VCR which is controlled by CABER.

CABER has been used in a range of different domains, mainly in the area of sports (Patrick and Lowdon, 1986; Patrick and McKenna, 1986a and 1986b). It has been developed on a Unix platform.

3.1.3 MacShapa

MacShapa (Sanderson, McNeese, and Zaff, 1994) has been developed to support scientific research in cognitive engineering. It is described by its authors as a “sequential statistical tool with a limited amount of model testing capability and a certain amount of multimedia capability” (Sanderson, Scott, Johnston, Mainzer, Watanabe, and James, 1994, p. 638).

MacShapa aims to integrate data from different data streams and to show the time relationships of the events within the different streams. The data streams can be in form of text strings, numeric values or codes, or can refer to external video data, replayed on a VCR. The data in MacShapa are presented in form of a spreadsheet. The columns of the spreadsheet are the different data streams, called the variables, the rows of the spreadsheet refer to the time. The user can see the data in the spreadsheet view, in a timeline view, which arranges data of selected data streams along a timeline, or in a listing, which is a compact representation of selected data from the spreadsheet. MacShapa supports the annotation and coding of video data. It controls a VCR and can time stamp and locate video segments. The code variables can be nominal, matrix or predicate variables. Nominal variables are words or combinations of words which are treated as one code. Matrix variables define a fixed sequence of nominal codes where each nominal code relates to a specific attribute of the behaviour to be described. Predicate variables consist of a main descriptive term, the ‘predicate’ followed by arguments specific to the predicate. For variables of type matrix or predicate MacShapa presents a form template in the relevant spreadsheet cells to assist in the data input. The data collected in the spreadsheet are stored in a database. MacShapa features a query language which consists of a condition - action pair. The condition has Boolean, relational, arithmetic, temporal or substring match operators. The actions are either database manipulation operators or output operators such as print, sum and count. A query can be applied to a selected range of spreadsheet cells.

MacShapa has been implemented for the Macintosh. Reference to a number of projects using MacShapa is presented in Sanderson (1998) and a review of MacShapa is given in Arnold, Kuk and Ritter (1995). Additional information about theories behind MacShapa can be found in Sanderson and Fisher (1994).

3.1.4 vPrism

vPrism (WCER, 1998) is a Macintosh system developed to support educational research. It works with digital video recordings and textual transcripts which have to be prepared with a word processor in a special format outside of vPrism. The user of vPrism sets up projects which can be divided into several units. Once video recordings and their transcripts are selected for analysis in a unit the user synchronises video recording and transcript. This is done by associating points of time in the video with the appropriate paragraph in the transcript.

vPrism provides six types of coding events. The event types define how codes will relate to video and/or text documents. The options are point of time and period of time and application to video only, transcript only, or both video and transcript. The user defines a coding event by giving a short character based identifier, a name, a description, and the event type. To perform the coding a coding event is selected and linked to video and text.

Searching in vPrism is possible for text strings or codes. A text string search applies to the text of the transcripts while a search for codes looks for instances where a coding event has been linked to a video or text segment. Searches can be performed within a project or can be restricted to single units. Based on the search results a video play list can be constructed. This list contains a selection of video segments which can be played in sequence. If the video play list is stored in the special format of a ‘video capsule’, it can be played by a PC based system called VideoVisor or vVisor.

3.1.5 Constellations

Constellations (Goldman-Segall, Marcovici and Halff, 1993; MERLin, 1998) aims to be a tool for the collaborative analysis of ethnographic multimedia data. It works with digital video and audio files, text files and images.

Working with Constellations centres around working with ‘pieces’ of video data. The researcher divides a video file into segments by specifying the start and end time of the segment, called piece of data. The pieces of data are coded with information about topic and participants in the video segment. These code instances can be given a numerical rating which is used later to judge the significance of the piece of data regarding a certain analysis topic (Goldman-Segall, 1993). The code information, called ‘tags’ in

Constellations, and the ratings are stored in a database. Tags, ratings and pieces of data together form a ‘star’. The user can attach a textual comment or part of a written transcript to a star. The user can query the information stored in the database to select specific stars. Several of these stars can be grouped to form ‘constellations’. Members of a constellation can be viewed in parallel. Collaborative data analysis is supported by identifying which researcher has produced which analysis data. Data produced by different researchers are stored in a shared database. Each researcher can retrieve other researchers’ tags and ratings and can view comparative charts for the data stored for a piece of data.

Constellations has been developed for Apple Macintosh computers. Information about the background of Constellations from an ethnographic research direction is available in Goldman-Segall (1993 and 1995).

3.1.6 CVideo

CVideo (Psychology Software News, 1995) is a system which by itself constitutes only a basic system to support the analysis of behaviour but can be used in context with other systems like NUDIST (see next paragraph). CVideo allows the user to control an external video recorder. It provides a special word processor which can be used to take notes and to link a specific video segment to these notes. A find command allows the user to search for keywords or phrases within the notes and the video controller of CVideo locates the appropriate video segment on the tape. CVideo can be used from within a range of Macintosh applications (personal communication, Jeremy Roschelle, June 1998).

3.1.7 NUDIST and CVideo

NUDIST (Richards and Richards, 1991; QSR Software, 1998) is a qualitative analysis system designed for the analysis of text. It can be used in conjunction with CVideo to support the analysis of behaviour captured on video. NUDIST allows the user to work with internal and external documents. Internal documents have to be in text format. External documents can have any format. For external documents NUDIST is restricted to storing the references to segments in the documents without being able to display these segments. Video data can only be treated as external documents in NUDIST where CVideo can assist in locating the document segments. Despite this weak support for the

analysis of video recordings it is interesting to look at the support NUDIST provides for the qualitative analysis of textual data.

In NUDIST the coding of data can create an ‘index tree’. The index tree combines two functions. It builds an hierarchical coding scheme and contains the code categories and their references to segments in the internal or external documents. A node of an index tree can either just define the code category or can contain the code instances as well. The code categories are entered as character strings. The code categories do not have a formalised structure. The structure of the coding scheme comes from the organisation of code categories into the index tree structure. The code instances can be produced in three different ways. The user can select segments in the documents to relate them to a code category. The user can perform a text search on the document to automatically code segments in the document. The user can perform a query on an existing index tree and apply a new code to the retrieved coding instances. This last way of producing codes is referred to as ‘system closure’ in the NUDIST literature (Richards and Richards, 1991). NUDIST provides a range of possibilities to query the index trees. It supports Boolean queries and relational queries relating to the segment positions of the code instances. Documents and index trees are stored separately in NUDIST for a number of reasons. The separate storage allows NUDIST to support references to external documents. Being detached from the documents, the index trees are very flexible structures which can be modified. Several index trees can refer to the same document. The documents can be viewed in their original form, not modified by the indexing process. The flexibility of the index trees is an important design criterion in NUDIST. The user is encouraged to change the index trees with the changing understanding of the data contained in the documents. Besides capturing analysis information by coding, the user can attach memos containing additional information or notes to documents, code categories and code instances. With its flexible index system, rich query options, memoing facilities and system closure NUDIST is designed to support the emerging of theories from the data. Reviews of NUDIST can be found in Weitzman and Miles (1995) and Walsh and Lavalli (1996).

A new program, NVivo (Richards, 1999; Fraser, 1999), which is based on NUDIST, has been released in mid 1999. NVivo, like NUDIST, focuses on the analysis of text documents. In comparison to NUDIST it allows the user to work with fully formatted as compared to ASCII texts and provides improved access to documents in other than text

format. The user can create links, for example, to digital video files. These files are still treated as external documents but now can be replayed on the computer using tools provided by the computer system executing NVivo.

3.1.8 ATLAS.ti

ATLAS.ti is a qualitative analysis system which was originally developed for the analysis of textual data (Muhr, 1991). In its newer versions it has been extended to work with video, audio, and graphics data but still focuses mainly on the analysis of text documents (ATLAS.ti, 1999). ATLAS.ti aims at theory building by supporting the exploration of data and the uncovering of phenomena.

To work with ATLAS.ti the user defines projects as 'hermeneutic units'. Each hermeneutic unit can contain several primary documents, which are of text, graphics, video, or audio format, and analysis data such as codes and memos. The user can define segments in the primary documents and these segments are called 'quotations'. A quotation can be linked to codes or memos. Codes are short text strings with an optional description. Codes can carry references to other codes resulting in conceptual networks. Memos are usually longer units of text and can have a user definable type such as 'theoretical' or 'descriptive'. Groups of primary documents, codes and memos can be built according to common attributes. These groups are called 'families'. Another grouping mechanism are 'networks'. In a network, primarily codes, as well as quotations, memos or primary documents, are linked to express semantic relations and to support theory building. The querying of codes, called 'semantic retrieval' in ATLAS.ti, can make use of the relationships expressed in a network. For text search, applied to primary texts, complex search macros can be defined and stored for later use. Beside being able to direct any output to printer, screen or file, ATLAS.ti has two other output formats. ATLAS.ti can generate output in the correct format for the statistical analysis package SPSS to perform quantitative analysis procedures. It can further produce output in Html format for display on the world wide web.

The current version of ATLAS.ti has been developed for the Windows NT platform. Reviews of ATLAS.ti can be found in Weitzman and Miles (1995) and Walsh and Lavall (1996).

3.1.9 Code-A-Text

Code-A-Text (Cartwright, 1998a and 1998b) is a system which has been designed for the analysis of data in text form and has later been extended to support the analysis of audio and video data. Code-A-Text claims to support qualitative and quantitative methods of analysis.

The strength of Code-A-Text lies in its features to analyse text. The system automatically creates an archive of all words occurring in the text to be analysed. The user can group these words to build up dictionaries of related words. Code-A-Text automatically creates indices according to the grouping of the words in the dictionary. Beside the automatically generated codes for text references, the user can describe text, audio and video data using content codes, categorical scales, numerical scales, open ended comments and direct annotations. Code-A-Text offers Boolean search on codes. The search output can be used to refer to the related data segments in text, audio or video format, to calculate descriptive statistics or to export the results in a range of formats. The search output can further be used to automatically apply new codes to the selected data segments.

Code-A-Text has been developed for the Windows platform and supports digital audio and video replay.

3.1.10 FERAL

FERAL (Carter, 1997; Carter and Patrick, 1997; Carter, Patrick, and Deane, 1998) was developed in the context of psychotherapy research. It was used for the combined analysis of psychodrama sessions captured on video and transcripts of expert commentaries relating to these psychodrama sessions. FERAL aims to support qualitative analysis and training in an open domain like psychotherapy.

FERAL is designed to provide the user with a very flexible environment to explore data in the form of video recordings and text documents. The user codes and annotates data and enters memos. The codes are created as needed during the analysis process. There are no formal requirements for codes, codes are text strings of unlimited length. FERAL supports text search in codes, code instances, annotations, and memos and searching for time codes. The interface design of FERAL follows the ‘sheet-of-paper’ metaphor. Like a sheet of paper on a desk all objects on the screen can be freely arranged to suit the

analysis needs of the user. FERAL works with popup menus to bring data and operations defined on these data close together.

FERAL has been implemented for Apple Macintosh computers using Hypercard. It controls a VCR via a serial port and displays the video picture on a separate monitor.

3.1.11 Other Systems

There are references to a range of other applications to support the analysis of data stored on video. These applications are either research tools or commercial applications. The descriptions of most of the research tools mentioned in the following paragraphs are, considering this fast moving area of information systems and computing, relatively old. A number of these descriptions concentrate more on technical aspects, like linking a description with a time code on a video tape or controlling a VCR through the computer, than on conceptional aspects, like the coding scheme, the coding procedures or the analysis possibilities. Attempts have been made to contact the authors of the referenced publications with the request for information on newer developments. Where these requests have been successful information about the current status of the tools has been included. For the commercial applications, the applications for which no academic publications could be found, even less conceptual information is available. Despite the lack of conceptual or up-to-date information it is still valuable to mention these systems and the publications behind them as they indicate the need for advanced video analysis tools in a wide range of application domains.

Action Recorder

Action Recorder (Psychology Software News, 1995) is an observation and analysis tool developed for an Apple Macintosh computer with Hypercard 2.2. It supports the recording of behavioural events, seen as indicators of behavioural states and changes in these states. The user assigns keys on a standard keyboard to the events belonging to a stream of behaviour. The recorded behaviours of the different streams are integrated into a single continuous record. This record can be queried for the behaviours of one single stream or for combinations of behaviours across several streams. The query results are formatted as command files for input into the SPSS statistics system. The concept of Action Recorder has not changed since 1995 (Watts, 1998). The use of Action Recorder is described in Monk and Watts (1995) and Watts and Monk (1996).

CAMERA

CAMERA (Vlugt, Kruk, Erp, and Geuze, 1992) is a system for collecting and editing ethnological data. Its design goals are to improve accuracy, reliability and training standards in the coding of behaviour. CAMERA controls an external VCR. The user defines codes for the behavioural events to be observed which are entered via a separate keyboard. To support the accuracy of the coding process CAMERA gives auditory feedback for pressing the keys on the keyboard and displays a mnemonic label for each code on the video monitor. The different coding passes are stored in traces. These traces can be merged or compared. CAMERA calculates elementary statistics on the coded events. An event plot shows the time properties of one or several traces. A review of CAMERA can be found in Hoogland (1994).

CCAS

The computerised coaching analysis system (CCAS) was designed to support systematic observation in a sporting environment concentrating specifically on the behaviours of coaches and athletes during a team sport practise (Franks, Johnson, and Sinclair, 1988). CCAS is used for live recording of behavioural events displayed by coaches and athletes during training sessions. The events are recorded using a touch sensitive digitisation tablet. There is only a brief mentioning of the use of CCAS in conjunction with video technology.

DVC 2000

DVC 2000 (The Dominion, 1998) is a commercially available video analysis system developed by Orad Hi-tec Systems in Israel. It runs on Silicon Graphics workstations and uses digital video. It allows linking text to video clips, to filter text and to sort results in a spreadsheet style document, and to replay selected video clips.

EVA: An Experimental Video Annotator

EVA (Mackay, 1989; Mackay and Davenport, 1998; Harrison and Baecker, 1992) has its origin in the area of usability studies. It aims to support the analysis of a range of related data sources being video recordings, transcriptions, keystroke logs and commentaries. EVA was developed for a Unix platform with X Windows. It uses digital video. Video sequences can be either annotated or marked with tags for further analysis. The user defines codes for certain events and uses these to tag specific video segments. The user

can create a snapshot of the current video image. This snapshot tag can be used to locate a video sequence. In the analysis phase the user views the video together with the tagged information and the annotations. The tags comprise a multimedia edit log which allows the user to view the related data in the different media formats all at once.

Hyperresearch

Hyperresearch (Hesse-Biber, Dupuis, and Kinder, 1991) is a HyperCard application developed for Macintosh computers. It supports working with a variety of source materials, online text and graphics documents, and external video and audio documents. Hyperresearch is designed to support qualitative data analysis. It is organised in three logical components, the coding component, the searching and reporting component and the hypothesis testing component. The user defines codes which can be applied to the source materials of the different media types. Hyperresearch presents the user with a code list, which is a list of all codes which have been used so far. Based on this list the user can modify single codes. The system offers automatic coding of occurrences of key words in text documents. The searching and reporting component supports searching for the occurrence of single codes or for Boolean combinations of codes. The search results can be saved as the current subset of code instances to be used for further analysis, can be used to create reports and can be used for calculation of basic statistics. The hypothesis testing component supports the automation of theory generation and theory testing. The user expresses theories in form of production rules. These production rules have the form ‘if expression then action’. They are tested on the basis of presence or absence of codes in the coding instances of source materials, which build the knowledge base. To use the hypothesis testing component efficiently the user has to include directionality into the coding, that is to not just code for a certain topic, like ‘self image’, but to further qualify it, like ‘negative self image’ and ‘positive self image’. The expressions of the production rules are built by Boolean combinations of codes from the code list. The actions of the productions are lists of instructions to add or remove codes from the knowledge base. The user can test or generate theories using multiple production rules.

KIT

KIT (Skou, 1998) has been designed for the qualitative analysis of recorded interviews and speeches. KIT has been implemented for Windows PCs and works with digital audio files. The user codes the audio recordings, can selectively replay audio segments, can

export audio data or summary data for statistical analysis, and can export notes for processing in a word processor. KIT does not work with video data.

Marquee

Marquee (Weber and Poon, 1994) is described as a tool for real-time video logging. The publication on Marquee concentrates on the development of the user interface for Marquee. Marquee supports pen-based input of personal notes and keywords in real-time. The notes and keywords are related to a time code on the video recording. The user creates time zones, writes notes and labels these with keywords. The keywords are used to link information in different notes. Because the notes are written by hand and not typed via the keyboard an automated searching through these notes is not easily possible. The hand written input was chosen to support the non-computer based note taking style of the potential users of Marquee. It allows not only the writing of words but also the drawing of symbols and sketches. The described implementation of Marquee focuses on the input mechanisms and offers only simple methods to access the logged information.

METACoder

METACoder (Ryan and Russell, 1994) was developed for PCs running a Windows environment. It controls a VCR with the video picture being displayed either on a separate monitor or on the computer screen. The suggested use for METACoder is in the areas of fieldwork situations, human and animal laboratory settings, teaching systems and market research. The user defines a number of events to be timed and/or counted. Recording of the events during the observation occurs via the mouse or the numeric keypad. METACoder calculates descriptive statistics. Data produced in METACoder can be exported for use in spreadsheet, word processing and statistics packages.

Observer

The system Observer (Hoiem and Sullivan, 1994) (a different system from the OBSERVER system mentioned previously) was developed in the context of usability studies. It was developed for Apple Macintosh computers using Supercard. Observer controls a VCR via a serial connection. The user defines a hierarchical set of codes and can use this to formulate coding sentences by selecting codes from different levels of the hierarchy. Additionally, the user can annotate code instances. Code instances are stored as ASCII files. Observer is used in conjunction with the systems Tracker and Reviewer

(both designed by the authors of Observer) and Microsoft Excel. In a usability study, two forms of data are collected. These are event data, like keystrokes or mouse movements, and behavioural data. Tracker is designed to capture the event data to complement the behavioural data collected using Observer. Reviewer synchronises the analysis of the behavioural and event data recorded with Observer and Tracker. Analysis data can be imported into Microsoft Excel for further analysis. Working with Observer and Reviewer, the user can produce a tape of selected video segments for presentation.

PROCODER

PROCODER (Tapp, Walden, 1993) is designed for the observation and coding of behavioural events recorded on video. It was originally developed to obtain data on infant social referencing. PROCODER runs on a PC and controls an external VCR. To develop the coding scheme the user can define codes of three different types: codes for events with an onset and offset time, codes used in designating intervals, and codes with a rating. The user can segment a video tape to indicate video material belonging to different sessions or to different elements within a session. For each segment the range of available codes can be specified. PROCODER uses this information to check if the entered codes are allowed for the current position of the video tape. It further checks for the input of correct codes. PROCODER checks for interobserver reliability and calculates descriptive statistics. Statistical data can be exported in ASCII format. PROCODER is still available and is currently undergoing redevelopment for a Windows95/NT platform with export options to a relational database. Conceptually PROCODER does not seem to have changed since 1993 (personal communication, Jon Tapp, June 98).

VANNA

VANNA (Harrison and Baecker, 1992; Harrison and Chignell, 1998; VANNA, 1998) was originally created to support video data collection and analysis for both software usability testing and behavioural studies in Computer Supported Cooperative Work (CSCW). It runs on Apple Macintosh computers as a HyperCard stack. It controls an external VCR. The video picture is presented on the computer monitor. VANNA emphasises real-time description of behaviour but additionally supports detailed analysis and exploration. VANNA supports coding for instantaneous events and for events which have a duration. For each event the user creates a button labelled with an appropriate name for the event. The buttons are displayed on the screen and have a different appearance for the two

different types of events. The user can enter textual comments either by themselves or in conjunction with an event. VANNA offers keyword searching for event instances and textual comments. The system provides descriptive statistics and a graphical representation of events on a timeline. Analysis results can be exported as ASCII files in formats suitable for import in applications like word processors, spreadsheets or statistical packages. In the design of VANNA specific emphasis was put on the design of the user interface. The system provides auditory and visual feedback during the coding of events. The user can customise the screen layout and can choose between multiple input devices.

VIDEOLOGGER

VIDEOLOGGER (Krauss, Morrel-Samuels, and Hochberg, 1988) is a system developed for an Apple II computer. It was designed for applications in behaviour analysis where videotaped information has to be quantified. The user can record events which are written onto the audio track of the video tape.

VideoNoter

VideoNoter (Roschelle and Goldman, 1991) is a system that was developed for Apple Macintosh computers. It can be used with either a VCR or a video disk player. VideoNoter was developed in the context of multi disciplinary analysis of behaviour and has, as well, been used for presentation and training. VideoNoter focuses on the analysis of one video recording from multiple perspectives. The users can define several ‘panes’. Each pane contains information for a specific analysis purpose or direction. The information can be textual annotations, transcripts or graphics, for example, to depict gestures. The panes can be viewed in parallel with the information in the single panes related to the time in the video they refer to. VideoNoter does not formalise a coding scheme, all textual information in the panes is unstructured text. VideoNoter can retrieve specific segments of video relating to a time or data in a pane.

3.2 Discussion of Features of Existing Systems

A number of key observations can be made by looking at existing systems to support the analysis of behaviour captured on video. These cover the topics of origin of the system, focus on analysis of video data, use of video technology, description mechanisms,

retrieval of information from description data, storage of analysis data, and reporting of results. Again, the focus here is more on conceptual than on implementation issues.

3.2.1 Origin of the System

There are three main areas the systems come from (Sanderson and Fisher, 1994):

- Behavioural research with a focus on quantitative analysis (OBSERVER, CABER, Action Recorder, CAMERA, METACoder, PROCODER)
- Social science research with a focus on qualitative analysis (Constellations, NUDIST, ATLAS.ti, Code-A-Text, FERAL, Hyperresearch, KIT, Marquee)
- Cognitive research in usability studies (MacShapa, EVA, Observer, VANNA)
(vPrism and VideoNoter cannot be clearly attached to any one of these areas.)

One thing these research areas have in common is that they make use of video (and audio) to capture behavioural data to be analysed. To analyse these video data there are a number of common requirements across the research areas like easy access to the video sequences, facilities to describe the behaviour displayed on the video, facilities to analyse the descriptions, and support for the analysis of several data documents of possibly different media types. Later sections of this work will look at these common analysis requirements in detail and will argue that these common requirements form the basis for the design of a system which supports the analysis requirements in all three research areas.

Besides the common requirements, each of these research areas has specific needs for the support of the analysis. These specific needs, originating from quantitative, qualitative and cognitive research methods, will be highlighted in later paragraphs.

3.2.2 Focus on Analysis of Video Data

The existing systems can be divided into two major groups:

- The systems which were designed primarily for the analysis of video data (OBSERVER, CABER, MacShapa, vPrism, Constellations, CVideo, Action Recorder, CAMERA, DVC 2000, EVA, Marquee, METACoder, Observer, PROCODER, VANNA, VideoNoter) or which at least included video as one of the media formats in the initial design (FERAL, Hyperresearch).

- The systems which were developed with a focus on the analysis of text documents and have been later extended for the analysis of video data (NUDIST with CVideo, ATLAS.ti, Code-A-Text).

3.2.3 Use of Video Technology

There are three different ways in which a computer system can interface with video data:

- Analog video is played on an external VCR. The system controls the VCR via a connection such as a serial port. The video picture is displayed on a monitor separate from the computer monitor. This solution allows the system to read the time code or frame counter from the VCR, to attach these to descriptions of the behaviour displayed on the video and to control the VCR to position the video to a specific location in the tape. The following systems have implemented this solution: CABER, MacShapa, CVideo, FERAL, Action Recorder, CAMERA, Hyperresearch, Marquee, Observer, PROCODER, VideoNoter.
- The system controls an external VCR as described above but the video picture is displayed in a window on the computer monitor. With this solution the output produced by the system and the video picture are displayed on the same monitor. VANNA and METACoder use this approach.
- The system uses digital video. The video data are stored on the computer's hard disk (or come from a digital video disk) and are played by a software video player. This means that no mechanical video player, a VCR, is involved. The video picture is displayed on the computer monitor. Systems using digital video are: ATLAS.ti, OBSERVER, vPrism, Constellations, Code-A-Text, DVC 2000, EVA.

The advantages and disadvantages of these different approaches will be discussed in Sections 3.3.1 and 4.1.

3.2.4 Description Mechanisms

Most of the systems provide two ways to describe the behaviour captured on the video. Firstly, they support a free-form, textual description which is commonly called ‘annotation’. Secondly, they provide a formalised form of description which in most systems is called ‘coding’ (or ‘indexing’ in NUDIST or ‘encoding’ in FERAL and MacShapa) with the data resulting from the process of coding called ‘code instances’.

Both of these types of descriptions are attached to a segment of the data displaying the behaviour. For textual data these segments can be, depending on the system, words, lines of text or any string of characters as selected by the user. For video documents a segment is a period of video with a start and end point. If start and end points have the same value then the duration of this period can be zero. Start and end points can be specified in a time unit or as frame counter values.

The setup of the formal description mechanisms, the coding, varies among the systems in definition of the codes, the possibilities for combining the codes, the possibilities for applying the codes to the behavioural data, the ways to store the resulting code instances, the flexibility to change existing codes, the range of queries which can be applied to the code instances and the information which can be extracted from the code instances. All this information related to coding will be referred to as ‘coding scheme’. For some of the mentioned systems the literature does not contain enough information to describe the coding scheme in detail.

In the following paragraphs the coding schemes of OBSERVER, NUDIST and CABER will be discussed. The coding schemes of these systems show the principal types of coding schemes available in today’s systems. These coding schemes are strongly influenced by the research tradition the systems come from. The weaknesses and strength of the coding schemes will be highlighted in Section 4.3.

The Coding Scheme of OBSERVER

The coding scheme of the OBSERVER clearly shows the system’s origin in quantitative behaviour analysis. Before the coding can start, the user defines the coding parameters and the codes. The user has to specify sampling and timing methods (for an explanation of these methods see for example Martin and Pear, 1992). The user defines independent variables, subjects, behavioural elements and modifiers. Independent variables describe an aspect of a behaviour which stays constant during the observation. These variables can be of character or numeric type and their value can be specified before, during or after an observation. This value is attached to the dependent observational data and can be used to filter data files for the analysis.

The subjects are the actors or receivers of behaviour. Observations can be single subject observations, in which case the actor does not have to be recorded for every observation,

or multiple subject observations with the actor recorded for every observation. The behavioural elements are the basic behaviours to be observed. The user specifies a behavioural element either as an event or as a state. Behavioural elements can be grouped into classes of mutually exclusive categories. In one class only one state can be active at a time, which means that the code instances for the states of one class can not overlap. To record several dimensions of a behaviour the user defines multiple classes. A combination of a subject and a specific class of a behavioural element is called a channel. For each behavioural element up to two modifier classes can be specified. The modifiers add information to the description of a behaviour.

Depending on the definitions of the subjects and behavioural elements, the coding sentences produced in OBSERVER have the form of ‘subject - behaviour - modifier1 - modifier2’ or just ‘behaviour - modifier1 - modifier2’. The subject is the actor in the described behaviour, the behaviour is the activity and the modifiers add more information to the description. The role of the modifiers in terms of the grammatical structure of the coding sentence is not specified in OBSERVER. Modifiers can have the role of objects, adverbs or adjectives. For each subject, behavioural element or modifier, the user can define a short key combination. The user can enter further descriptions for the mentioned codes.

More detailed information about the OBSERVER coding scheme can be found in Noldus Information Technology (1998a). To summarise, the coding scheme of the OBSERVER can be characterised as follows:

- Definition of codes before start of the coding process;
- Definition of specific properties of the codes (as states or events);
- Definition of groups of codes (classes) and combinations of codes (behavioural elements and their modifiers);
- Design of the coding scheme to facilitate the continuous description of a specific behaviour (states grouped into classes);
- Design of the coding scheme to facilitate the parallel description of several distinct aspects of a behaviour (channels);
- Definition of keycodes for faster data entry.

The Coding Scheme of Nudist

Coding in NUDIST has a very different approach to coding in OBSERVER. In NUDIST, following its background in a qualitative research tradition, the user starts working through the behavioural data and defines the codes (or indices as they are called in NUDIST) only as the need arises out of the data. NUDIST defines no parameters, like sampling or time methods, for its codes; it defines no rules how to combine codes into coding sentences. A code itself can be a description of a specific aspect of a behaviour, a thought or comment of the user analysing the behavioural data, a one word code or a whole sentence. Working with NUDIST the user creates an index system consisting of hierarchical trees constructed by codes and code instances and of additional information in the form of memos. The hierarchical trees can be modified during the course of a analysis project.

In summary, the coding scheme in NUDIST can be characterised as follows:

- Definition of codes (indices) as needed in the analysis process;
- Development of an index system based on hierarchical trees;
- Emphasis on the flexibility of the coding scheme.

The Coding Scheme of CABER

The system CABER has a unique approach to the definition of a coding scheme. In CABER the emphasis lies on capturing the interactions of multiple actors and their activities. Before a project in a specific behavioural domain can be started using CABER, a coding language has to be defined for this domain.

This coding language is specified as a formal language. The single codes are the terminals and non-terminals of this language. The grammar rules define the coding sentences which can be constructed within the coding language. Due to the nature of a formal language the correctness of each coding sentence can be tested. CABER will accept only grammatically correct coding sentences. This test for correctness ensures that only descriptions of behaviour can be entered which have been included in the design of the coding language. The grammar of a CABER system will usually be defined to produce descriptions of a form like '*who did what to whom using what*'. The grammar will in detail specify which actors can perform which activities and which additional parts of information can or have to be recorded. The nature and variation of coding sentences in a

CABER system depends on the specification of the grammar. Once a grammar has been defined for a domain and the user has started to collect behavioural descriptions, the grammar cannot be modified without losing access to these descriptions.

The key points of the CABER coding scheme are:

- Specification of a grammar to define a formal language on behavioural domain basis;
- Emphasis on correct coding sentences which capture the relationships of actors and their activities;
- Grammar cannot be changed during the lifetime of a study.

The other systems introduced in earlier sections have coding schemes which are in principle either like the OBSERVER or the NUDIST coding schemes or which combine certain elements of these coding schemes. In cases where elements of these coding schemes contribute to the ideas of this research these will be referred to in the appropriate paragraphs.

3.2.5 Retrieval of Information from Description Data

The way information is retrieved from the description data depends on the format of these data. For annotations, which are free-form textual descriptions, the mechanism for selection of specific annotations is text search. For code instances the selection mechanism depends strongly on the underlying coding scheme. The selected descriptions are used in four different ways:

- To function as pointer to a specific location in the behavioural data;
- For the calculation of quantitative measures;
- To deduce information from the descriptions themselves;
- To establish relationships within the data.

In particular in systems from a qualitative background, like NUDIST or FERAL, text search cannot only be applied to annotations but also to codes, memos or comments. The codes in these systems are usually not single words but combinations of words or clauses. The descriptive information is contained in the codes themselves rather than in the combinations of codes. Therefore it is of value to perform a text search within each code. Across all systems the results of a text search are used to extract information from the textual descriptions themselves and to study the behavioural data segments the

descriptions are linked to. In text oriented systems like NUDIST and Code-A-Text text search can additionally be used to automatically code the textual behavioural data. All systems provide the facilities to query the code instances for single codes. Most systems provide options to query for combinations of codes.

OBSERVER and CABER make use of the structures for combining codes into coding sentences as defined in their coding schemes. In OBSERVER the user can query for specific actor - behaviour - modifier sequences. In CABER the exact query possibilities depend on the definition of the domain specific grammar. Following the design ideas behind CABER, the user is likely to ask queries regarding the relationships of actors, their actions and the recipients and objects of these actions.

Most systems allow queries using Boolean or relational combinations of codes. The Boolean queries are constructed with AND, OR, and NOT operators. The relational queries ask for time or position sequences in the code instances. Depending on the design of the system, the emphasis and possibilities in analysing sequences of code instances are different. In a qualitative system like NUDIST the user can identify code instances which are overlapping or in close proximity to each other. In systems like OBSERVER, CABER or MacShapa the emphasis is on strict sequences of code instances.

An additional way to extract information from the code instances comes from a qualitative system like Hyperresearch. Hyperresearch facilitates hypothesis testing and does that by supporting rules in the form of ‘if Expression then Action’. The expression has the form of a Boolean query applied to the code instances and the actions are instructions to add or remove code instances. As well, coming from a qualitative background, ATLAS.ti allows the use of relationships between codes captured in a semantic network.

Systems from a quantitative behaviour analysis background, like OBSERVER, Action Recorder, CAMERA, PROCODER, and, to a certain degree systems from a usability testing background, like MacShapa and VANNA, make use of the design of their coding schemes which allows the specification of behavioural events with and without duration. Based on these specifications the systems can calculate durations and frequencies of events.

NUDIST uses the structure of its index trees to query for code instances belonging to a specific index tree or to a specific branch of an index tree.

The different options for retrieving information from the descriptions of the behavioural data can be summarised as follows:

- Text search applied to annotations;
- Querying for single codes;
- Querying for combinations of codes as defined in a coding language;
- Querying for Boolean combinations of codes;
- Querying for code sequences;
- Querying for codes using the properties defined for these codes;
- Querying for codes using the hierarchical structures defined for these codes.

3.2.6 Storage of Analysis Data

Of the mentioned systems only Constellations uses a database management system, a relational database, to store and manage the analysis data. The other systems store the data either in ASCII files or in data files with a system specific format. Most of the systems support the further analysis of data in applications like word processing, database, spreadsheet, statistics, graphics or model building by providing export formats (OBSERVER, MacShapa, NUDIST, ATLAS.ti, Code-A-Text, Action Recorder, CAMERA, Hyperresearch, METACoder Observer, PROCODER, VANNA).

3.2.7 Reporting on Results

The systems provide several different ways to present results during the analysis process. All systems display the results of a text search or a query, usually in the form of a list. NUDIST provides graphical presentations of the index trees. Systems like OBSERVER, CAMERA, EVA, and VANNA offer graphical presentations of events along a time line. In OBSERVER, vPrism, and Observer the user can compile a video play list to easily replay selected sequences of video segments. ATLAS.ti can export data in Html format.

As mentioned above, most systems provide export facilities to other applications. Reports on the overall findings of a study are then constructed in these applications.

3.3 Areas for Improvement

The study of the concepts and implementations of existing systems to support the study of behaviour suggests a range of possible improvements for the design of a new system. These are listed below and subsequently discussed.

- Most of the systems described use analog video technology. The improved quality of digital video makes it interesting to look at analysis possibilities which come with the use of digital video or multimedia in general.
- A number of the systems have not been designed as multimedia systems but concentrate on the analysis of data of one media type.
- Even some of the systems that claim to support quantitative and qualitative analysis clearly favour one of the analysis styles and offer little support for the other analysis style.
- The coding schemes offered by the systems provide limited expressiveness considering the complexity of behaviour.
- Systems in the area of study of behaviour have to deal with a wide range of data of different types: behaviour recordings, description data, analysis data. Yet nearly none of the systems uses a database system to manage these data.
- Systems of the kind described have to be highly interactive which therefore demands systems designed according to best HCI principles.
- The systems should accompany the whole process of a behaviour analysis study, from preparation to analysis and reporting on the findings of the study. The current systems concentrate mainly on the analysis and therefore leave room for improvement in preparation and presentation of findings.

While this chapter outlines the areas for improvements Chapters 4, 5 and 6 introduce detailed suggestions for these improvements.

3.3.1 Technology

The first step to use computer systems to assist the analysis of data captured on video was to control the VCR via a serial port connection to the computer. This allowed exchange of accurate timing information between the computer and the VCR and consequently led to automatic positioning of the video tape. While this technique is a great improvement over having to manually control the VCR, it still has the disadvantages of analog video. Non-

sequential access to the video data is very slow and video tapes suffer relatively quickly from a loss of quality (Petty, 1987). Most of the systems using analog video display the video data on a separate monitor. This has the disadvantage that the user has to divide their attention between the video and the computer monitor. To bring behavioural and analysis data closer together, some systems display the video pictures in a window on the computer monitor while still using analog video.

A number of systems are already using digital video. Digital video eliminates the disadvantages of analog video, slow access and wear. Digital video does not degrade in quality, regardless of how often it is played. Access to any video segment is instant because no mechanical video player is involved and the video, regarding its access properties, is no longer a sequential medium. All systems using digital video display the video picture on the computer monitor.

The replay quality of digital video on a multimedia PC with video compression card is already very good. These PCs can replay video in full frame rate, full picture size and good quality. The challenge for a new information system to support the analysis of video data lies in extending the analysis possibilities for video data. Current systems using digital video have mainly mapped the functionality of a conventional, analog VCR into a digital video player. The systems make use of the random access facilities and greater flexibility in replay speeds of the video but they have not explored the possibilities of using digital video any further.

3.3.2 Multimedia

Most of the current systems are multimedia systems in the sense that they facilitate the analysis of behavioural data in text and video (and for some systems in audio or image) format. The systems allow one to code and annotate behavioural data segments and to retrieve these segments according to their descriptions. The access to data of multiple data types has become better due to the use of digital technology for audio and video.

While the use of multiple data formats for input data is valuable, the existing approaches could be complemented by producing multimedia output data to present the findings of a study as combinations of text, video, audio and image documents. A number of the current systems have not originally been designed as multimedia systems, but have either

concentrated on the analysis of text documents (like Code-A-Text) or have only access to analog video (see Section 3.2.3). A challenge for a new concept for a system, which is specifically designed as a multimedia system, is to explore which well-known analysis techniques can be transferred from one data medium to another and how the analysis of several input data streams, of different or same media types, can be coordinated.

3.3.3 Combined Quantitative and Qualitative Analysis Systems

Some of the systems introduced claim to support both quantitative and qualitative analysis needs. More quantitative systems also offer qualitative tools such as annotation and memoing (like OBSERVER) and more qualitative systems offer quantitative elements like descriptive statistics (like Code-A-Text). After presenting arguments from the literature for combining quantitative and qualitative analysis methods in Section 2.3, ideas for a closer combination of the methods will be introduced in Chapter 4.

3.3.4 Coding Schemes

The discussions in Section 3.2.4 have introduced three different types of coding schemes: the typical quantitative coding scheme like the OBSERVER's, the typical qualitative like NUDIST's, and the grammar based coding scheme of CABER.

In the study of human behaviour it is often necessary to describe the interaction of several persons or the interaction of persons with objects. Examples for this are the studies on violence in television systems (Watson et al., 1991), team sports (Patrick and McKenna, 1986b; Patrick and Lowdon, 1986), or the use of remote meeting technologies (Ruhleder and Jordan, 1998). The study on violence in television systems looks at the perpetrators and victims of violence, the type of violence, the weapon used, the context of the violent act, and the way this act is presented in the television system. The study collects the information 'who did what to whom, where, using what'. The analysis of team sports deals with a large number of players. The performance of a player can only be judged in the context of the support this player received from their team mates and the attention they received from the opposing team. Questions like 'how often did the player receive a pass?' or 'when did the player lose a ball after being tackled by an opposing team member?' might be asked. To be able to answer these questions during the analysis process the behaviour descriptions must contain information about the interaction of

several individuals. The study on the use of remote meeting technologies looks at the behaviour of the members of two groups in remote locations. The interactions within each group and between the groups are studied.

The three types of coding systems described have different approaches to capture complex patterns of behaviour. In the quantitative scheme each group of codes focuses on a single aspect of the overall behaviour. The coding can either be done by parallel coding for several single aspects of behaviour or by coding in multiple passes with each pass concentrating on a single aspect of the behaviour. The resulting code instances are collections of single codes relating to the same behavioural data. They are only connected through their references to time or position information in these data. They are not related through features of the coding system like combining several codes to form a coding sentence. (The combination of codes to ‘actor - behaviour - modifier1 - modifier2’ in OBSERVER is an exception. Here codes are combined into a sentence of ‘subject - verb – object’ format. Yet the role of the object in the sentence is not specified at any greater depth and no greater complexity of sentence patterns is possible.) Descriptions of a complex behaviour have to be constructed from several descriptions of simple behaviours. In a domain with a high density of behaviours by several actors it can be impossible to relate the correct actor - action - object combinations to each other by looking just at the time or position information.

In a qualitative coding scheme, as well as in a quantitative scheme, there is no provision made to combine single codes into coding sentences. In comparison to a quantitative scheme a code in the qualitative scheme contains a range of information in itself. The qualitative scheme aims at retrieving information by providing detailed information within each code rather than through the combination of codes. New codes are defined on an ‘as-needed’ basis during the analysis. The richness of single codes in a qualitative coding scheme can, for example, be seen in the listing of codes presented in Carter’s study of psychodrama (Carter, 1997). In the study Carter used codes like ‘dir encourages awareness that group members part of system’ or ‘dir works to bring out nature of tele links’ (Carter, 1997 p. 244; ‘dir’ stands for the director in the psychodrama session).

Two problems arise with this approach to coding in qualitative coding schemes. The first problem is the potentially large number of codes which can be defined. Because

descriptions of behaviour cannot be constructed by combining several codes, even slight variations in the descriptions make it necessary to define a new code. With a high number of codes used in a study, it can become increasingly difficult to extract information regarding a specific property of a behaviour where the description of this property might be contained in several codes. The following example highlights this point. Assuming that the description of a behaviour involves two participants (p1, p2) who each might perform three possible actions (a1, a2, a3), already six different codes have to be defined (p1 a1, p1 a2, p1 a3, p2 a1, p2 a2, p2 a3) to construct only very simple descriptions of behaviour. With an increasing number of participants and actions and a more detailed description of behaviour, the number of codes necessary increases rapidly.

The second problem arising from the currently common qualitative coding schemes is that there is no way to semantically correctly retrieve partial information from within a code. The codes in a qualitative coding scheme are defined simply as text strings. A computer system has no mechanism to detect meaning in these text strings, different from the human user who interprets the characters of the text string as words of a sentence. If the user now attempts to retrieve partial information from the code instances ('select all code instances which involve the director (dir)'), the only mechanism for a qualitative analysis system to do so is to apply a text search to the code instances. While this text search will retrieve all code instances containing the string 'dir', the system has no possibility of semantically interpreting the coding sentence. Looking for more complex patterns of behaviour, this can lead to either retrieving semantically invalid code instances ('false hits') or retrieving an incomplete set of code instances ('false miss'). This problem is illustrated in more detail in Section 4.3.1.

The grammar based coding scheme of CABER puts strong emphasis on the combination of codes into coding sentences. This results in semantically rich coding sentences with a clearly defined syntax. The problem identified for qualitative coding schemes, where the semantic content of a coding sentence is not accessible to the computer system, therefore, does not exist in a grammar based coding scheme.

The problem of the grammar based coding scheme lies in the need to define the grammar before the start of the study. This requires an intimate knowledge of the behavioural domain and a clear understanding of which coding sentences will be required in the study.

Because the semantic information contained in the coding sentences is only accessible using the information built into the grammar, it is problematic to change the grammar and, therefore, the range of possible coding sentences once the coding has started.

Chapter 5 will introduce a new coding scheme which retains the strength of the grammar based coding scheme while avoiding the problems associated with it.

3.3.5 Database Management System

With the exception of Constellations none of the systems uses a database management system. The systems store their analysis data in ASCII or data files. The internal organisations of these files are system specific. In several system descriptions the authors refer to the database of the system. These databases do not have the functionality of database management systems in storing, organising and managing the data as described in the information systems literature (Date, 1995; Silberschatz, Korth and Sudarshan, 1997).

Behaviour analysis studies potentially have to deal with a large amount and variety of data. There are behaviour recordings, supporting documents, descriptions of behaviour in general, descriptions grouped by certain analysis criteria and results. Where behaviour is observed over a longer time frame, multiple streams of data have to be stored, analysed and compared (Martin and Pear, 1992). Data are analysed several times by the same criteria to calculate inter- and intra-observer reliability (Martin and Pear, 1992). The same data are analysed in different ways to explore different theories. Complex coding schemes are designed and could be re-used in a different study of the same domain.

To cope with these challenges a system to support the analysis of behaviour should make use of a database management system which is suggested in Section 4.2.

3.3.6 Interface Design

A number of issues in interface design have already been addressed by some of the systems. FERAL allows the definition of codes in an unlimited length of characters. Some earlier systems did restrict the number of characters available for a code and therefore limited the expressiveness of the code names (Weitzman and Miles, 1995). CABER uses a key tablet for the input of codes in the coding process. The use of a key tablet speeds up

the entry of codes. Carter (1997), in the development of FERAL, puts strong emphasis on the flexibility of the screen layout, which he characterises as a ‘sheets-of-paper’ metaphor. The user can choose which windows are visible and which position and size they have on the screen. The move to a flexible positioning of windows can as well be seen in the development of Constellations, which in earlier versions had a fixed arrangement of several sets of related data in one window and which has moved to a flexible arrangement of related data in several windows (MERLin, 1998).

Chapter 6 will introduce another element of interface design, the management of windows grouped according to their function in the behaviour analysis process. With the increased functionality of a system to support the analysis of behaviour, as suggested in this work, the number of parallel or overlapping tasks the user performs, resulting in an increased number of interface elements, requires the system to provide assistance to the user in managing the interface elements.

3.3.7 Support for the Whole Behaviour Analysis Process

Current systems concentrate on the analysis phase of a study. They provide no explicit support for the combination of several studies of one domain neither do they integrate the analysis results of the studies to form a repository of findings as a valuable source of knowledge for future reference.

Chapter 4 will make proposals about combining the data, created to prepare a study, and the conclusions, achieved in a study, to a collection of domain specific information available for use and reference in future studies.

3.4 Conclusion

In this chapter existing computer systems to support the study of behaviour were reviewed. Firstly, the systems were introduced individually by describing their most prominent features. Secondly, the systems were compared with each other with the aim of identifying common conceptual features. This comparison focused on the origin of the systems in terms of the disciplines involved in the study of behaviour, on the analysis of video data and the use of video technology, on description mechanisms and the related retrieval facilities, on the storage of analysis results, and on the reporting of study results. Based on this comparison a number of areas were identified in which a new system could

provide advanced support for the study of behaviour. The use of digital video and multimedia in general could provide advanced analysis and presentation possibilities. A new system could be explicitly designed to support both quantitative and qualitative analysis methods. A new coding mechanism could be developed to allow for the description of complex behaviours and an automated retrieval of these descriptions. The design of a system could be improved by using a database management system and by applying HCI principles.

The next chapter, Chapter 4, builds on the review of the existing systems presented here and on the study of analysis methods as described in Chapter 2. It introduces, on a generic level, the conceptualisation of an advanced multimedia information system to support the study of behaviour.

Chapter 4

Conceptualisation of Advanced Multimedia Information Systems

In the previous chapter the areas in which existing the programs for the support of the study of behaviour could be improved were identified as use of technology and multimedia, combined quantitative and qualitative analysis systems, coding schemes, database management systems, interface design and support for the whole behaviour analysis process. The conceptualisation presented in this chapter addresses these areas. This is done taking into account the requirements of the analysis methods applied in the disciplines involved in the study of behaviour (see Chapter 2):

- Behaviour is commonly recorded in video, audio, text and image format;
- Both quantitative and qualitative analysis methods are applied;
- The analysis process consists of the three core elements *description*, *retrieval* and *interpretation*;
- The analysis process is concluded by the reporting on results.

Based on the review it can be suggested that an information system has to provide support for the following phases of the analysis process:

- Preparation;
- Description;
- Retrieval;
- Interpretation;
- Conclusion.

Preparation

At the outset of a study, a number of steps have to be undertaken to prepare the analysis. Study goals have to be defined, data collection methods specified and coding schemes prepared. In terms of an information system to support the analysis of behaviour recordings, the preparation phase includes aspects like the identification of computer files containing behaviour recordings, the naming and description of the study, and the input of relevant parameters of the coding scheme.

Description

In the literature review in Chapter 2 it has been established that two forms of descriptions are commonly used in studies of behaviour. *Annotation* refers to a description in natural language form while *coding* implies a formalised classification following the definition of a coding scheme. In the following discussion the term *description* is used to refer to both *annotation* and *coding*. The outcome of a description process is a number of *description instances*. Each description instance consists of a *description sentence*, the textual description of the behaviour, linked to a *segment of a behaviour recording*, which contains a representation of the behaviour described.

Retrieval

Following on from the description, the next phase in the analysis process is the *retrieval* of descriptions according to some common characteristic or theme. The term *retrieval* is used here to summarise what is commonly called *searching*, applied to annotations, or *querying*, applied to code instances. While the format of the retrieval depends on the type of description, the motivation for the retrieval step is independent thereof. The task of this phase in the analysis process is to extract descriptions either based on a common characteristic in the description sentences or on some properties relating to the associated segments of behaviour recordings. These extracted descriptions are referred to as *thematic groups of description instances*.

Interpretation

In the interpretation phase the behaviour analysts work with the description instances which have been retrieved according to specific characteristics. Depending on the research method applied, the analysts will focus on different aspects of these thematic groups of description instances. They will interpret either the individual description sentences, sequences of description sentences, the segments of behaviour recordings or numeric values derived from the description instances.

Conclusion

To conclude the analysis process the findings of the study have to be formulated. These findings can be illustrated by data used or produced during the analysis. The term *conclusion* is used to refer to the outcome of a study. A *conclusion* is seen as consisting of the two elements of *conclusion statement*, the formulation of the findings, and *conclusion data*, any material supporting the findings.

The information system conceptualised in this chapter is modelled based on these phases of the study process. In context of the information system the term *analysis* is used to comprise the phases *description*, *retrieval* and *interpretation* which leads to the specification

of an information system consisting of the three major components Preparation, Analysis and Conclusion, called **PAC**. A graphical overview of the phases in the behaviour analysis process and the data resulting from each phase is given in Figure 4.1.

In the following sections features of an advanced multimedia system to support the study of behaviour are suggested. The first two sections consider technical aspects of such a system while the remaining sections look at features arising from needs of the domains involved in the study of behaviour.

4.1 Use of Multimedia Technology

The availability of powerful multimedia computers opens up a wide range of new opportunities for the use of multimedia data (Lennon, 1997). The possibilities for supporting the study of behaviour are outlined in the following paragraphs:

- Integration of behaviour recordings of multiple media types;
- Support of common timelines;
- Transfer of analysis techniques from paper based to electronic material;
- Video specific analysis tools;
- Use of multimedia data to support conclusions.

Integration of behaviour recordings of multiple media types

Using digital multimedia technology, data of the different media types, video, audio, text and image can all be accessed on the same computer hardware device. It is, therefore, possible to combine the use of behaviour recordings of different media types within one analysis system.

Data of multiple media types have different properties and can be used in a wide variety of ways which has implications for the concept development:

- A behaviour can be recorded in multiple media formats to allow for the analysis of different aspects of the behaviour.
- The data can constitute reference material, like an electronic copy of a book, to compare the behaviour recordings against.
- A segment of a recording can be a time based sequence in case of video or audio data, a character sequence for a text document, or an area of an image.
- The position information relating to a behaviour recording can be a time value in the case of video or audio data, a character position for a text document, or a pair of pixel coordinates for an image document.

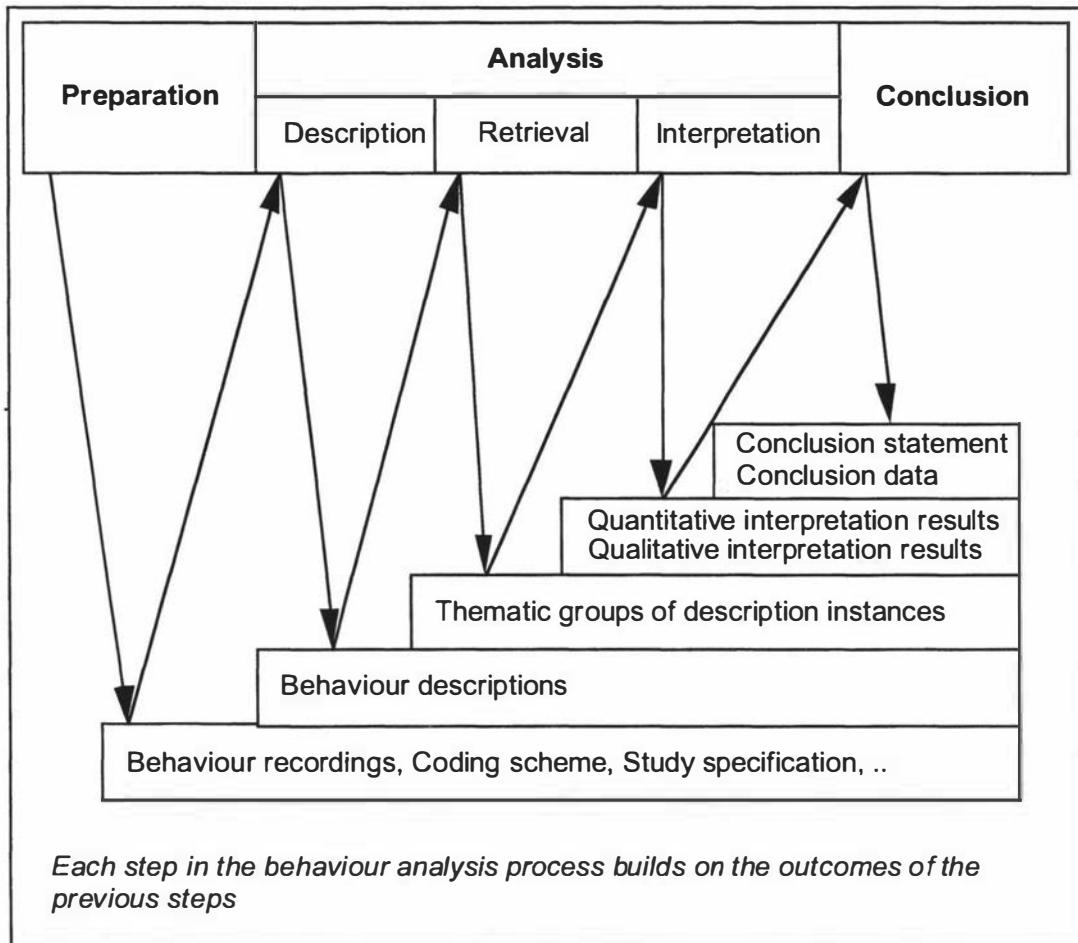


Figure 4.1: Phases in the behaviour analysis process

Analysing behaviour recordings of multiple media types has implications for the retrieval of behaviour descriptions based on the position of the information of interest in the recording segment attached to these descriptions. This will be discussed in Section 5.2.3.

Support of common timelines

If a behavioural setting contains a group of people or moving subjects, it is often not sufficient to record the behaviour from just one direction. Carter (1997), for example, uses three camera perspectives for the parallel recording of the actors of a psychodrama session. Nearly every sports recording for television presentation is performed with several cameras to be able to capture the action from the best perspective.

If the same behavioural sequence is recorded in parallel from different viewpoints, or with differently positioned microphones, the resulting behaviour recordings share a common

timeline. Conceptualising a *multimedia* information system it is possible to take advantage of the parallel recordings in several ways:

- A multimedia system can replay several behaviour recordings at the same time. This can involve showing several video clips in parallel, presenting video clips and audio recordings in parallel, or playing a number of audio recordings concurrently. Conventionally this effect is achieved by combining up to four video recordings on one video tape using special hardware equipment, as it has been done in Carter (1997). While this allows several recordings of different viewpoints to be looked at in parallel, it lacks the flexibility which can be achieved with a multimedia system. A multimedia system can offer the user a choice as to which of the parallel sequences are to be presented in a physical arrangement suitable for the current analysis. The user can instantly change between viewing parallel sequences and concentrate on the most appropriate recorded camera angle for the current analysis focus.
- Besides the pure presentation of parallel sequences the analysis tools of the information system can reflect the parallel nature of the recordings. If suitable for the current analysis, a description attached to one of the parallel sequences can automatically be referenced to all other parallel sequences. This feature could be used initially to describe a specific behaviour by looking at a recording of the behavioural setting showing a general overview. After retrieving the descriptions for the behaviour, the analyst could replay the behaviour sequences from the parallel recording showing a specific viewpoint. An example for this analysis technique can be taken from a team sport like soccer. By looking at the overview recording of the game attempted shots at goal could be identified. For the detailed analysis of these game situations, the analyst might want to study the parallel recordings of cameras showing the goal area and the perspective of the goal keeper. By utilising the parallel nature of the behaviour recordings, the analyst only has to describe the behaviour presented in one video recording but has immediate access to the appropriate sequences of the other parallel recordings.

Transfer of analysis techniques from paper based to electronic material

Working with paper copies of text documents, like transcripts of interviews, analysts use tools like coloured marker pens to highlight interesting passages or to scribble comments across the text. As described in contributions to an electronic discussion list for qualitative researchers (Carter and Weitzman, personal communication, July 1996), many analysts prefer to work on paper compared to electronic data material. While some of the aspects of paper, like ‘the feel of paper’, can hardly be replicated working with a computer system other aspects can be included in the design of a multimedia system.

For electronic text documents it is possible to replicate ‘highlighting’ and ‘scribble’ analysis techniques which are described for paper material for example in Krathwohl (1997). The user could highlight text elements, without being forced to follow a strict character sequence as it is required in common word processing packages. A search functionality could allow searching for text sections containing highlighting of a specific colour. A scribble function could be implemented similarly, allowing the user ‘to write across’ the electronic text.

Using highlighting and scribbling tools for video recordings presents the additional challenge of time coordinating the user input with the moving video picture and leads to the design of video specific analysis tools.

Video specific analysis tools

Digital video can be accessed in a far more flexible way than analog video. One major advantage of digital video is the possibility of random access to individual video frames. This eliminates rewind or fast forward operations and makes the reviewing of single sequences much more efficient. Having the ability of positioning the digital video exactly on a frame or to a precise time saves time in finding a specific scene. A multimedia system can be configured to replay a digital video at any replay speed, from extreme slow motion to fast browsing through the recording. Additional functions like a selection of the first or last couple of frames of a video clip or a selection of a limited number of frames taken in regular intervals throughout the clip can be implemented. An analyst could use such functions to obtain a quick impression of the contents of the video recording. A series of consecutive frames would allow for the detailed analysis of a particular movement.

Following on from the idea of highlighting electronic documents, a range of video specific annotation tools can be designed. These tools are thought to produce overlays over the video images without modifying the video images themselves (much like a transparency can be put on top of an image). Highlighting aspects of a behavioural sequence on top of a moving picture can be used to direct the attention of the viewer to a specific component of the behaviour. This technique is commonly used in the presentation of sports on television, for example to indicate in which direction a basketball player will be moving and to assist the sports reporter in explaining the following sequence of action to the viewer.

Another way to use highlighting on video is to assist the analysis of a specific image or sequence of images. Tools like grids, concentric circles or lines can be used to judge distances or the relative position of objects to each other as presented in the images. Examples of the use of these tools can again be taken from the reporting of sports on television or in videos containing teaching material. In the game of bowls it is often very hard to judge the distance of the balls to the centre. Looking at the game from a birds eye

perspective concentric circles can be drawn around the centre ball. Following the concentric lines it is very easy to compare the distances of the different balls to the centre ball and therefore to analyse the game situation. In a movement like a golf swing, the position of several body axes in the different phases of the swing is very important. Analysing a golf swing, it could be useful to look at a sequence of single frames and to overlay the images with lines, representing the different body axes. Having the lines on top of the images could assist in determining the correct or incorrect relationship of the axes. This technique could be used for the analysis of the swing by a golf trainer and could as well assist the trainer in explaining the deficiencies of the swing to the golf learner.

Use of multimedia data to support conclusions

A multimedia system combines behaviour recordings of different media types in one system. During the analysis process the behaviour analyst identifies sequences in the behaviour recordings which show key elements of the behaviour. Additionally, the analyst produces data in various formats like written comments, graphs or tables showing numeric results, graphs displaying concept relationships, or behavioural data sequences graphically annotated. Because all these data are available within the multimedia information system, they can be used to illustrate the findings of the study. The analyst can select the data which best support the conclusion and integrate these into the conclusion presentation.

The conclusion statement itself can be formulated in different media formats. An audio recording of a conclusion statement could be accompanied by the presentation of image sequences to highlight aspects of the study findings. A textual conclusion statement could be presented in parallel to video or audio sequences from the behavioural data.

4.2 Use of a Database Management System

An information system to assist the analysis of behaviour recordings, as it is conceptualised here, has to deal with a large variety of data (see Section 3.3.5):

- the ‘input’ data in the form of behaviour recordings, and additional data representing domain-wide or study-specific information;
- the ‘working’ data as description, retrieval, and interpretation data;
- the ‘output’ data which represent the findings of the study.

Beside dealing with the different types of data, a behavioural study is likely to produce several repetitions of data of one data type:

- additional behaviour recordings are included to study a behaviour over an extended time frame;

- several observers perform the same types of behaviour descriptions;
- alternative directions of analysis are pursued.

A single study can belong to a range of related studies in a behavioural domain with the advantage of referring to data used in previous studies:

- behaviour recordings or description structures set up for one particular study can be used in subsequent studies;
- analysis data or conclusions from one study can be referred to in another study.

The amount of data produced and the nature of processes applied to these data in a behavioural study strongly suggest the use of a database management system. Several authors describe the benefits of using database management systems (Date, 1995; Silberschatz, Korth and Sudarshan, 1997). Silberschatz et al. (1997, pp. 2-3) mention the following data management problems which can be addressed by using a database management system: data redundancy and inconsistency, difficulty in accessing data, data isolation, integrity problems, atomicity problems, concurrent-access anomalies, security problems.

Applied to an information system to support the analysis of behaviour recordings these problems, which can be overcome by using a database management system, are illustrated as follows:

- *data redundancy and inconsistency*: some data, like names for behaviour recordings or data sets, have to be provided by the users of the application; it has to be ensured that different users do not define different names for the same data;
- *difficulty in accessing data and data isolation*: looking at the definition of a sophisticated coding language (as defined in Section 5.1) sufficient data structures and data management have to be provided to facilitate description and retrieval of code instances; retrieved data might be used in different ways, depending on the type of interpretation performed (as outlined in Section 4.4);
- *integrity problems*: different analysis directions within one study can be based on common analysis data, such as descriptions; changes to analysis data by one analyst have to be kept separate from changes performed by other analysts; elements of the coding scheme, for example, might have to be modified during the lifetime of a study and it will be necessary to manage multiple versions of a coding scheme;
- *atomicity problems*: removing a behaviour recording from the system might require a series of subsequent delete operations to prevent data inconsistency;
- *concurrent-access anomalies*: should several analysts work on the same study and use the application concurrently data inconsistencies could occur;

- *security problems*: behaviour recordings and data resulting from their analysis are usually subject to confidentiality and access to these data has to be controlled.

4.3 Need for Flexible Structured Coding and Query Languages

There are two principal ways to formulate the descriptions of the behavioural data. One is the use of natural language in the form of textual input where the resulting descriptions are commonly called annotations (or memos for longer textual descriptions)(Weitzman and Miles, 1995). The other is to use a formalised coding scheme where the resulting descriptions are called code instances and the process of description is called coding (Weitzman and Miles, 1995).

In this section the strengths and weaknesses of both types of description languages, the natural language and the formalised coding scheme, as they are used in current multimedia systems, are highlighted. The new Flexible Structured Coding Language, FSCL, which attempts to combine the strengths of both existing approaches while at the same time avoiding their weaknesses, is introduced in Section 5.1.

The mechanisms to extract information from the behaviour descriptions are closely related to the format of these descriptions. Text search is applied to textual descriptions while queries are applied to code instances (Weitzman and Miles, 1995). The Flexible Structured Query Language, FSQL, to accompany FSCL is specified in Section 5.2.

4.3.1 Natural Language for the Description of Behaviour

The strengths of the use of natural language as a description language are its expressiveness and flexibility, its immediate availability to every analyst with access to a word processing facility, and the familiarity with and understanding of natural language for every analyst. This is reinforced by the emphasis on textual description mechanisms to support the analysis process (Weitzman and Miles, 1995; Sanderson and Fisher, 1997).

The weakness of natural language as a description language becomes only apparent if one wants to make use of a computer system to analyse the descriptions. Natural language with its lexical ambiguity, ambiguous sentence structure, and context dependency of meaning is very complex (Smeaton, 1997). Even if the meaning of a sentence is immediately apparent to a human reader, it can be extremely difficult for a computer system to detect this meaning. Good computer programs for natural language processing are very complex, specialised programs which are still far away from being able to analyse all sentences correctly (Smeaton, 1997). Lexical matching techniques may retrieve irrelevant or inaccurate results (Witter and Berry, 1998). The risk of retrieving descriptions of behaviour

with a different meaning from that intended might be acceptable if these descriptions are closely inspected by the analyst with incorrectly retrieved descriptions being discarded. Where the retrieval of descriptions is used as the basis for the calculation of quantitative measurements and therefore the detailed inspection of each description is not part of the analysis process, the risk of retrieving descriptions with the ‘wrong’ meaning cannot be tolerated.

The current systems that support the study of behaviour rely on text search to assist in the analysis of natural language descriptions (NUDIST, FERAL, Code-A-Text). None of the systems uses natural language processing techniques, presumably because of their complexity. The problem with the text search approach regarding the meaning of text can be illustrated with the following example sentences.

Bill hit Jim and ran.

For the human reader it is obvious what this sentence describes: Bill is performing two activities, he hits and runs. Assuming one wishes to use a text search method to search a set of behavioural descriptions for the behaviour ‘Bill ran’. Searching for the character sequence ‘Bill ran’ will not get a match with the example sentence even though the sentence describes the behaviour requested. With this text search strategy one can achieve only an incomplete match with behaviour descriptions. Changing the text search strategy by including a wildcard operator into the text string and searching for the sequence ‘Bill*ran’ will retrieve the example sentence. Yet a new problem becomes obvious if one considers the following example sentence:

Bill hit Jim and Jim ran.

The application of the text search ‘Bill*ran’ to this sentence produces a match that would incorrectly identify the sentence with the behaviour ‘Bill ran’ when it is obvious to the human reader that this is invalid.

These examples show that applying a text search strategy cannot deal with the content of even simple natural language sentences.

4.3.2 Structured Languages for the Coding of Behaviour

For formalised coding schemes the strengths and weaknesses depend very much on the definition of the coding scheme and the coding process. Coding schemes, in current implementations of systems which support the analysis of recorded behaviour, usually allow for the definition of words or word combinations as codes and the combination of these codes into hierarchical structures (OBSERVER, NUDIST). In the coding process

these codes are applied to the behavioural data sequences either as single codes (NUDIST, FERAL) or in predefined, very restricted combinations of codes (OBSERVER).

These coding systems express meaning mainly within single codes (NUDIST, FERAL) and only in a limited way in the combination of codes (OBSERVER). They have the strengths of simplicity of use and relative flexibility for changing the coding scheme.

The restrictions of these coding schemes can be highlighted by considering the previous example sentences. To express meaning within one single code the whole sentence ‘Bill hit Jim and Jim ran’ could be introduced as one single code. To analyse the relationships within the code, which is built as a natural language sentence, the same reference problems are experienced as with natural language descriptions, as previously mentioned (see Section 4.3.1). If, instead, one defines codes for each individual element, like ‘Bill’, ‘Jim’, ‘hit’, ‘ran’ and ‘and’, more flexibility is gained by combining these codes into sentences. At the same time, one could not determine the relationships between these single codes and retrieval would suffer, again, from the same problems as in analysing natural language sentences using text search techniques.

A very different approach for the definition of a coding scheme was taken in CABER. Here, a domain specific formal language is defined which attempts to address all observable behaviours within this domain (Patrick, Ho and John, 1985). The resulting coding sentences produced with CABER contain rich meaning and, because the definition of the formal language considers the relationships between single codes, it can be readily accessed by a computer system. The disadvantages of this approach lie in the need to incorporate every observable behaviour in the specification of the formal language and the inflexibility of this formal language concerning extensions and modifications. A further disadvantage is the need for the coder or observer to learn the domain specific formal language.

4.3.3 Comparison of Natural Language Description and Formalised Coding

The consideration of natural language as a description language has shown that the weakness of this approach lies in the complexity of accessing the rich meaning in the description sentences in an automated way with the assistance of a computer system. The potential advantage of using a formalised description language lies in the possibility of extracting meaning in an automated way. From the observations on existing coding approaches, a table is presented here, indicating the features that are present or absent in the various approaches (Table 4.1). In the design of a coding approach expressiveness and flexibility are related to ease of computerised extraction.

	Natural Language (1)	Independent Codes (2)	Strict Grammar (3)
Expressiveness	+	-	+
Flexibility	+	+	-
Computerised extraction of meaning	-	-	+
<p>(1) <i>a description in natural language, either as annotation or as a single code containing a natural language sentence</i></p> <p>(2) <i>a non-formalised system of codes that can be combined in an entirely unrestricted manner</i></p> <p>(3) <i>a combination of codes to a coding sentence formalised via a grammar with a fixed vocabulary</i></p>			

Table 4.1: Comparison of different approaches to description

4.4 Interpretation Techniques

In the sequence of analysis steps, the interpretation follows the steps of description and retrieval (see introduction to Chapter 4). After the description of the recorded behaviours, thematic groups of description instances are extracted according to criteria ranging from properties of the description sentences to properties of the attached behaviour recording segments. The different interpretation techniques are then applied to the groups of description instances. An overview of the relationships between the different forms of description, retrieval and interpretation is given in Figure 4.2. The possible interpretation techniques focus on various aspects of the groups of descriptions, ranging from quantitative to qualitative approaches and can be characterised as *calculation, counting and comparison, as study of sequences of description sentences, as viewing of behaviour recording segments and as study of description sentences*.

Calculation, Counting and Comparison

This type of interpretation is based on the calculation of single measurements and the comparison of several of these measurements. Single measurements are:

- Counts of the number of description instances within one thematic group;
- Size of the behaviour recording segments attached to the description instances, where the measure of size depends on media type: duration for time based media like video

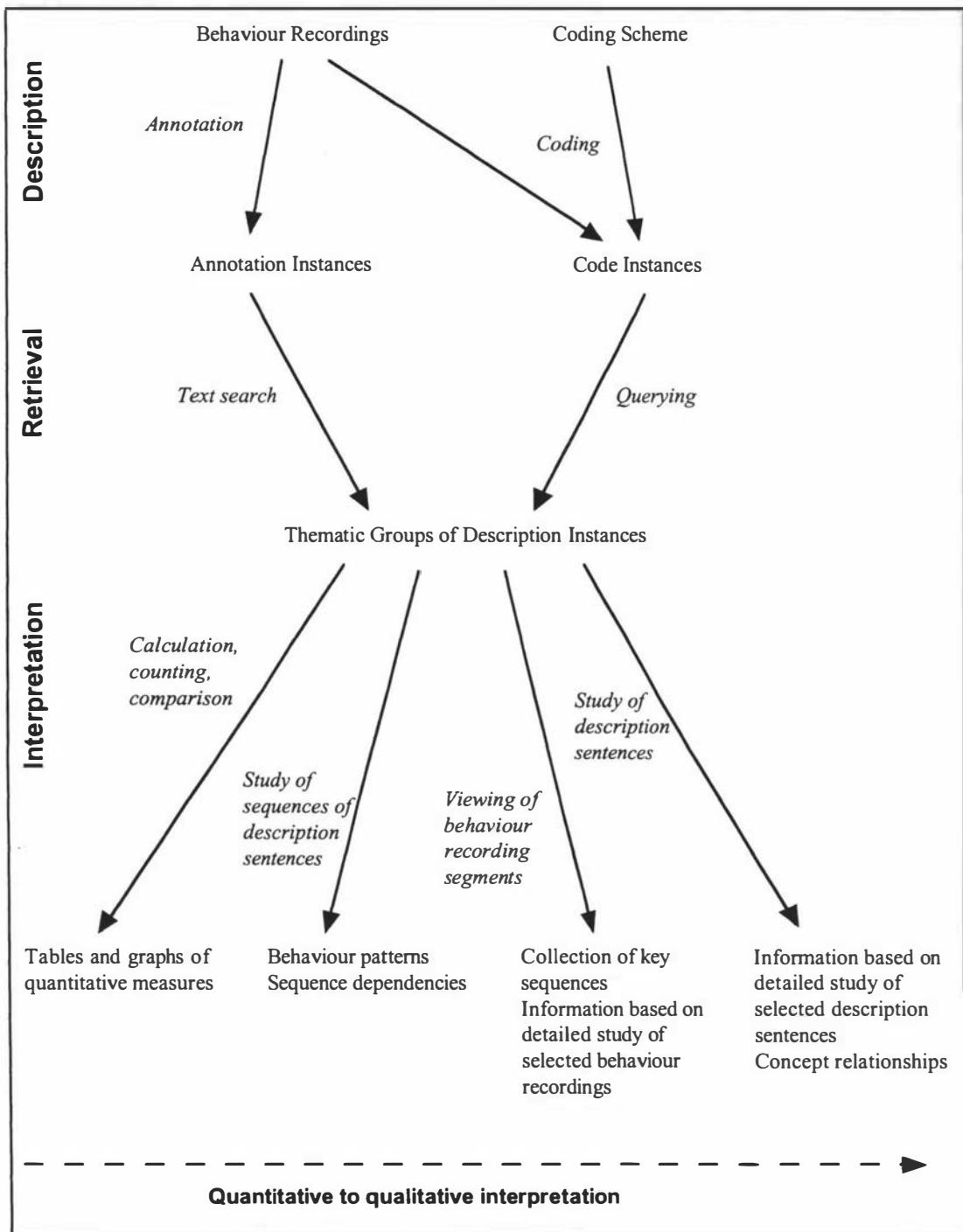


Figure 4.2: Relationships of description, retrieval and interpretation techniques

and audio; number of characters for text segments; size of image areas for image documents;

- Frequencies or percentage of coverage based on the total length or size of the behaviour recording.

These single measurements, calculated for several thematic groups of descriptions, can be compared with each other and presented as graphs or tables.

Study of Sequences of Description Sentences

The interpretation of sequences of description sentences occurs in two forms:

- Statistical sequential data analysis in form of Markov or lag sequential analysis as described for example in Gottman and Roy (1990). This type of interpretation requires a formal way of coding, an adequate sample of data and sampling techniques. It is used to provide information about dependencies between events (Sanderson and Fisher, 1997).
- Study of sequences of behaviour descriptions in a qualitative sense by analysing rich descriptions of related behaviour components (see the ‘Reading Groups’ case study in Section 7.2).

Viewing of Behaviour Recording Segments

Each description instance has attached to it a behaviour recording segment. The behaviour recording segments belonging to a thematic group can be used for interpretation in several ways:

- Study of single behaviour recording segments in detail; this can mean the study of movements, emphasis or intonation in spoken words, or reading of a text segment in context with the surrounding text;
- Study of single behaviour recording segments using specific replay features like slow or fast motion;
- Study of single behaviour recording segments applying graphical annotation tools like overlays or highlighting (as described in Section 4.1);
- Study of several behaviour recording segments in parallel to compare the development of a behaviour over time or to compare behaviours of different participants;
- Study of behaviour recording segments in combination with reference material, for example, to compare a participant’s movement to an ‘ideal’ movement.

Study of Description Sentences

Both types of description sentences, the annotation sentences and the coding sentences, can contain very rich and detailed descriptions of complex sequences or interactions of behaviour. Through the appropriate retrieval mechanism, text search or query, groups of

descriptions get selected based on some common characteristic. This characteristic can be rooted in either the text of the description sentence or in the behaviour recording segment attached to the description. In the interpretation phase the analyst can study the description sentences of the selected group of descriptions with all their details. This can lead to the discovery of concept relationships, which can be captured and used in subsequent retrievals.

The meaningful use of the above mentioned interpretation techniques has to be considered in the context of the whole design of the study. The calculation of correct quantitative measurements is only possible based on description and retrieval results which avoid problems like duplicate counts of identical description instances or time comparisons for data on different time lines. The specific requirements for quantitative interpretation in the context of behaviour recordings of multiple media types are addressed in Section 4.5. Qualitative analysis is rarely conducted in a strict linear fashion. The use of the qualitative interpretation techniques has to be seen in interaction with the steps of development of the coding scheme, description, retrieval and additional techniques like memoing.

Within the constraints of this research the interpretation techniques can only be described in a somewhat simplified form. A whole range of specialist applications are available for the mentioned types of interpretation. Specialised statistical packages are available, which offer, besides the basic descriptive statistics mentioned here, a range of tools for more advanced descriptive statistics, interpretive statistics and sequence analysis. Some specialised qualitative analysis applications offer a range of features for concept building and theory testing. Many of these applications are described in Weitzman and Miles (1995). Features of these applications could be integrated into the concept presented here.

4.5 Specific Requirements of Quantitative Interpretation in a Multimedia Context

The thematic groups of description instances arrived at through retrieval processes form the basis for the quantitative interpretation. One thematic group can contain description instances which can be linked to several behaviour recordings of possibly different media types. The range and format of the operations which can be applied to these description instances depends on the media types involved.

Operations to calculate durations and frequencies have to be adjusted depending on the media type of the behaviour recording segments they are applied to. For time based media duration is expressed in time intervals. For character based media or images the ‘duration’ operation has to be adapted to mean a collection of characters or image areas. Accordingly,

the frequency operation has to be adjusted to the right measurements of time, character or image area units.

If the description instances refer to behaviour recording segments of different behaviour recordings but same media types, the total duration can be calculated by adding the duration for each behaviour recording. For the calculation of frequencies it has to be taken into account that the basis for the calculations has to be the combined length of all behaviour recordings involved.

If the description instances refer to behaviour recording segments of different behaviour recordings of different media types, a calculation of duration or frequency cannot arrive at one single resulting value. Characters, time units and image areas cannot reasonably be combined in one calculation.

The above considerations indicate that the calculation of quantitative measurements can lead to meaningless or misleading results if behaviour recording segments of multiple behaviour recordings and possibly multiple media types are involved. A possible approach is that operations to calculate values like duration and frequency are only performed if all behaviour recording segments of the thematic group of descriptions refer to the same behaviour recording. Other issues affecting the correctness of quantitative measurements occur in the combination of data from multiple description and retrieval sets and are outlined in Section 6.4.

4.6 Support for Conclusion Construction

During the analysis process, analysis results of different formats are created. Quantitative measurements in the form of graphs or tables are produced. Significant behaviour recording segments are identified which might have been graphically annotated. Concept relationships have been discovered and captured. All these analysis data will have contributed to the findings of the study and carry important information which can be used to support and illustrate the study conclusion. A study conclusion can be regarded as consisting of two parts:

- The conclusion statement which formulates the findings of the study;
- The conclusion data which consist of data collected during the analysis that support and illustrate the findings.

Such a conclusion can be put together in the form of a multimedia presentation. The conclusion statement is either recorded as audio data or typed as text document. The conclusion data, which exist already from the analysis steps, are attached to the appropriate sequences of the conclusion statement. Replaying the conclusion creates the effect of the

analyst reporting the findings of the study while at suitable points of time during the presentation the supporting data are displayed. This multimedia presentation can be used within the information system to support the study of behaviour or can be exported in a format suitable for presentation on the world wide web with its inherent multimedia capabilities (see Chapter 8).

The exact format of a conclusion presentation has to be determined in context with the purpose of the study. A number of uses for a conclusion in multimedia format can be envisaged:

- In the context of academic research the conclusion presentation can be used to supplement information presented in an academic paper. The conclusion can be made available to the research community through the world wide web.
- In the areas of coaching or training the conclusion can be used by the student or learner. The conclusion can be studied independently from the presence of the coach or trainer, yet can explain and illustrate the shortcomings in a behaviour. For the example of a wrongly executed movement, the following elements can be recorded in a conclusion presentation. The coach explains the problem with the movement by showing some behaviour recording segments, indicating areas on the images where the movement was wrong. The coach presents a statistic showing that the wrong movement led to a number of mistakes to highlight the need for improving the movement. The following sequences of the conclusion recording compare the movement to an ‘ideal’ movement performed by the same study participant or an expert in the field. The presentation is concluded by training suggestions for working on the improvement of the movement. The participant studies the conclusion presentation repeatedly, independently from the presence of the coach. The coach prepares the presentation with the data material collected during the analysis of the movement.

An example for the use of video sequences to improve the performance of movements is the work on ‘video feedforward’ conducted by Dowrick (1991). Lee and Fielding (1991) suggest the building of, with the help of multimedia computing, simulated social environments used for the training of researchers.

- In a clinical setting, where the behaviour of a patient is assessed on a continuous basis, video recordings of the behaviour can be taken and analysed. A conclusion presentation can be recorded and stored for later access. Over time a set of conclusion recordings is built up, documenting the behaviour of the patient over time. Having the possibility to include video segments in this documentation allows the creation of a very rich description of the behaviour.

4.7 Domain Context and Knowledge Building

A study of behaviour will often be conducted not in isolation but in context with other studies in the same domain. Setting the study in a domain context will allow the analyst to reuse some of the data available to the information system:

- The analyst may be able to reuse parts of the coding scheme developed for previous studies. The coding scheme can be seen as being defined on a domain basis with a selection of specific elements used on a study basis.
- The analyst or a group of analysts can build up a collection of reference materials on a domain basis with each study referring to a suitable set of reference files.

The conclusions, with their statements and accompanying data, will be available for reference in further studies:

- A particular behaviour can be followed up over time. The analyst can refer to previous study results for the same behaviour.
- Conclusions from related previous studies can be looked at.
- Analysis steps from previous studies can be examined.

Over time and after conducting a number of studies of behaviour using the information system, a repository of data and knowledge could be built up for use by a group of analysts.

4.8 Conclusion

In this chapter the outline for a new multimedia information system to support the study of behaviour has been given. Such a system consists of the three major components *Preparation*, *Analysis* and *Conclusion*. On a generic level features of such a new system have been discussed considering possible contributions of the discipline of information systems and the requirements of the disciplines involved in the study of behaviour. The generic ideas, which have been outlined in this chapter, are followed up by a specification for a concrete system, the system PAC, in Chapter 6.

Before the specification for PAC is given, the following chapter, Chapter 5, concentrates on a central feature for the conceptualisation of improved information systems support for the study of behaviour. Based on the discussions of existing approaches for the description of behaviour, natural language annotation and coding, the new coding and query languages FSCL and FSQL are introduced.

Chapter 5

The Flexible Structured Coding and Query Languages

In Chapter 4 the advantages and disadvantages of natural language and formalised coding for the description of behaviour were discussed. Based on these discussions Chapter 5 introduces the Flexible Structured Coding Language, FSCL, and the Flexible Structured Query Language, FSQL. FSCL combines a flexible vocabulary with an explicit grammar. FSQL builds on the structures of FSCL and additionally considers the multimedia properties of the behaviour recordings attached to FSCL descriptions.

5.1 The Flexible Structured Coding Language (FSCL)

The aim of the development of FSCL was to create a coding language suitable for the description of behaviour. The range of sentences required to describe this behaviour was discussed with experts in the various fields of studies of behaviour. It was further influenced by looking at the types of sentences supported by systems like CABER and by looking at descriptions produced in studies like Carter's work on psychodrama (Carter, 1997).

The definition of two types of sentences resulted from this process:

- The description of activities performed by actors in sentences of the form *subject verb object* with example sentences like 'Mike hit ball' or 'Bill runs to teacher'.
- The expression of concepts or description of situations in partial sentences of the form *concept object* with example sentences like 'rain in interval' or 'attention Mike' where the construction of a full sentence could be more lengthy without adding information (as in 'Mike shows attention').

The design of FSCL was influenced by one further consideration. While it would not be possible 'to force' the user to enter meaningful descriptions of behaviour, the design of FSCL should ensure that the user could enter only sentences with a correct semantic structure. In this light it should be possible to construct sentences like 'attention Mike' to describe a situation but not sentences like 'Mike ball' which does not contain any information about the relationship between Mike and the ball.

5.1.1 Concept of FSCL

Any language consists of two main components: the vocabulary and the grammatical rules which define how to combine words of the vocabulary into meaningful sentences. The approach taken with the design of FSCL is to allow a flexible definition of the vocabulary and to provide a fixed, **generic grammar** for expression and extraction of meaning. The flexible definition of the vocabulary, that is the freedom to introduce any word at any time into the coding language, aims at providing a natural language like environment in which the user has the words available to describe any occurring behaviour. With the explicit definition of a grammar, which is much simpler than the implicit grammar of natural language, the coding language is provided with a means of combining words into meaningful sentences. At the same time, due to the relative simplicity of the grammar, it is able to automatically extract from the database meaningful sentences by computational techniques.

The element of FSCL which allows the combination of a flexible vocabulary with a fixed grammar structure is the introduction of **categories**. All words in the vocabulary belong to one of six categories and the grammar is defined on these categories. The grammar defines how the different elements of a sentence, like subject, verb, object, can be combined, whereas the categories define which words of the vocabulary can be used as subject, verb or object. By defining the grammar on categories and not on individual words it is possible to specify the grammar in a generic way. This means that the same grammar can be used across studies of different behavioural domains. The only modification needed to use the coding language for the description of new behaviours is in the definition of suitable vocabulary.

5.1.2 Categories of FSCL

The categories of FSCL serve the purpose of grouping the words of the vocabulary. The grammar rules of FSCL which define how to combine words into sentences are defined on these categories. By knowing which category a word belongs to, a computer program can determine in which way this particular word can be used within a sentence. New words can be introduced into the vocabulary without having to modify the grammar rules as these only apply to the categories which remain unchanged.

The categories of FSCL have been defined in accordance with the ‘word classes’ of the English grammar. Word classes have generally replaced the traditional categorisation of words into ‘parts of speech’ (Wardhaugh, 1995; Jackson, 1980). While there is some variation in the definition of the word classes the main classes are defined consistently by most authors as:

- nouns, pronouns, verbs, adjectives, adverbs, conjunctions and prepositions (Wardhaugh, 1995);
- nouns, (full-)verb, adjective, adverb, determiner, pronoun, preposition, conjunction, operator-verb, interjection and enumerator (Leech, Deuchar, and Hoogenraad, 1982);
- nouns, verbs, adjectives, adverbs, pronouns, numerals, determiners, prepositions and conjunctions (Jackson, 1980);
- nouns, verbs, adjectives, adverbs, prepositions, pronouns, conjunctions, determiners and auxiliary verbs (Freeborn, 1995).

Based on these word classes the following categories have been defined for FSCL:

P Person/Thing
A Activity
C Concept
K Conjunction
D Descriptor
R Preposition

The word class ‘nouns’ has been divided into the two categories of Person/Thing and Concept. This distinction was made to support the generation of the two sentence types as outlined in the introduction to Section 5.1. While words of both categories can function as objects in a sentence, only words of the category Person/Thing can be actors and only words of the category Concept can be used in the place of concepts in the *concept object* clauses. The word classes ‘adjectives’ and ‘adverbs’ have been combined into the category Descriptors, which also contains enumerators. These word classes were combined into one category to simplify the setting up of the vocabulary (see Section 5.1.3) and the grammar of FSCL (see Section 5.1.4.1). The word classes ‘verbs’, ‘conjunctions’, and ‘prepositions’ translate directly into the categories Activity, Conjunction, and Preposition.

The only word class, looking at the definition of Warghaugh (1995), which has no equivalent in a category is the word class ‘pronouns’. Pronouns have been omitted from the definition of FSCL to keep the analysis of a sentence relatively simple. Sentences containing pronouns can only be understood either in the context of other sentences or of clauses within a sentence. While pronouns play an important part in the use of natural language, it was considered that they could be omitted in the coding language because they can be replaced with the noun they stand for semantically. The following example sentence without a pronoun is not as natural as the sentence with a pronoun but can be easily understood by a human reader and can be more easily analysed by a computer system:

Jim ran to the slide and the teacher asked him to wait.

Jim ran to the slide and the teacher asked Jim to wait.

Examples of words in each category are given in the next section on the vocabulary of FSCL.

5.1.3 Vocabulary of FSCL

Any natural language word can be defined as a word of the coding language's vocabulary. The word has to be defined as belonging to one of the six categories of the coding language.

To increase the richness of description for a behaviour and the analysis of these descriptions, the words within one category can be combined into hierarchical structures. These hierarchical structures allow the grouping of words which have common properties. Using hierarchical structures, the coding of behaviour and the analysis of the resulting coding sentences can be on a detailed or on an aggregate level (Richards and Richards, 1995). Any word is allowed to belong to multiple hierarchical structures.

Besides using hierarchical structures to relate words to each other, words can be linked to other words by specifying word links. This allows the introduction of a 'second level' of combining words according to common properties.

Figure 5.1 shows some of the vocabulary definitions of an exemplar study of children's playground behaviour. The main focus while setting up this vocabulary was to group the children according to their membership of the different classes in the junior and senior sections of the school. For each student a link to the words 'Male Students' and 'Female Students' is set up. It is assumed that the coding of the playground behaviour is done on individual students resulting in coding sentences like 'Bill hit Jim and ran'. Besides extracting behavioural descriptions from the collected coding sentences on an individual student basis, the defined hierarchies and word links provide the following analysis options. Using the hierarchies one can search for descriptions involving junior and senior students in specific roles of behaviour as in 'senior student hit junior student'. Using the word links one can search for descriptions of behaviour focusing on a secondary property of the students in being either male or female students as in 'female student hit male student'. Both these types of queries can be applied to the same range of coding sentences because the specific information needed to evaluate the queries is contained in the definition of the vocabulary.

5.1.4 Formal Specification of FSCL

The main emphasis for developing the grammar for FSCL was to be able to correctly analyse each behavioural description into its components and to extract the correct relationships between these components. The sentences to be accepted by FSCL were

Category Person/Thing	Category Concept	Category Preposition
Students	School day	in
Senior Students	morning session	on
Bill	interval	under
Andy	afternoon session	from
Junior Students	Emotions	
Jim	happiness	
Clara	joy	Pace
Jenny	anger	slow
Students	frustration	medium paced
Female Students	Start	fast
Male Students	End	Intensity
Playground	Category Conjunction	soft
slide	and	medium
swing	or	strong
Teacher	before	
Category Activity	after	
Physical behaviour	<i>Hierarchy definition expressed through indent structure</i>	
run		
jump		
hit		
Verbal behaviour		
talk		
shout		
cry		
	<i>Word Link definition expressed through</i>	

Figure 5.1: Exemplar vocabulary definition

sentences of the forms *subject verb object* and *concept object* as introduced at the beginning of Section 5.1 and combinations of these clauses building simple and compound sentences. A simple sentence is defined as a sentence containing only one clause, called the main clause, which can have compound subjects or predicates. A compound sentence is built by two or more coordinated main clauses (Wardhaugh, 1995).

To simplify the description of possible sentences in FSCL the terms ‘clause’ and ‘predicate’ are introduced. A clause can either be a ‘subject verb object’ combination or a ‘concept object’ combination. A predicate is a ‘verb object’ combination.

To be able to describe more complex behaviours in FSCL the following ways of combining clauses and predicates into sentences are introduced:

- clause conjunction clause conjunction ... clause;
- predicate conjunction predicate conjunction ... predicate.

On a more detailed level, with the aim of providing the richness of a natural language like language, the following elements are allowed:

- any number of descriptors before words of the categories Person/Thing, Activity, Concept;
- prepositions as part of an object;
- two or more actors performing the same activity, where an actor is a word of the Person/Thing category.

With these language elements it is possible to formulate rich descriptions of behaviour and still to be able to analyse these descriptions correctly for their meaning, which will be shown in Section 5.2.1.

5.1.4.1 Automaton and LL2 Grammar

FSCL is specified in two parts. The first part of the specification is an automaton which is used to group the words of the coding sentences into structurally relevant components. The second part of the specification is defined as an LL2 grammar which is used for analysing the structure of the sentences (see Herschel, 1974, or Louden, 1997, for the definition of LL2 grammars).

The automaton is applied to the pattern strings of category identifiers. These pattern strings can be derived for each coding sentence by retrieving the category types for the words of the sentence from the vocabulary database, mimicking the process of lexical analysis for a compiler. The role of the first pass of the syntax analysis is to combine the words of the sentence into word groups with the aim of revealing the structural components of the sentence.

Words grouped together are Descriptors with the Person/Thing, Activity or Concept they describe and Prepositions with their Person/Thing or Concept word. This grouping has two effects. Firstly, it simplifies the definition of the syntax grammar by reducing the complexity of an unknown number of descriptors and by identifying word groups containing Prepositions as objects in the structure of a sentence. Secondly, by identifying the structural components of a sentence, the storage of a sentence in the form of a semantic tree (see Section 5.1.4.2) as an object is prepared.

Only the information about the structural components of a sentence is used for the second pass of the syntax analysis, yet the full information about all words belonging to a structural component is retained during the syntax checking process and later stored in the semantic tree.

The first pass of the syntax analysis produces a modified pattern string which identifies the word groups in the sentences. This modified pattern string then becomes the input for the

second pass of the syntax analysis using the LL2 grammar. The identifiers of the modified pattern string are:

- p Person/Thing with Descriptors
- a Activity with Descriptors
- c Concept with Descriptors
- r Prepositional Person/Thing or Concept with Descriptors
- k Conjunction

The automaton, together with its equivalent regular expression, is given in Figure 5.2. Table 5.1 presents the transition table for the automaton. An example of a coding sentence with its pattern and modified pattern strings can be found in Figure 5.3.

After grouping the words into structural components in the first pass of the syntax analysis the role of the second pass of the syntax analysis is to identify the relationships of these structural components to each other. These are the subject - verb - object relationships and their combinations with conjunctions plus the partial sentences in the form of concept - object. An LL-type grammar has been chosen for this part of the analysis because of the ease of implementation of the top-down syntax analysis considering the relatively small size of the grammar. The full LL2 grammar is presented in Figure 5.4.

5.1.4.2 Semantic Tree

When providing a description language producing coding sentences which can be queried for meaning, it is not sufficient to know that a coding sentence is syntactically correct. Besides the test for correctness of the coding sentence, it has to be stored in a representation which facilitates access to the semantic of the sentence. This representation is done in the form of a **semantic tree**.

The concept of a semantic tree is introduced as compared to a syntax tree. A syntax tree is a representation of all productions applied in analysing a sentence (Herschel, 1974). The format of a syntax tree to store the structure of dictionary entries has been used in Patrick, Zhang and Artola-Zubillaga (1999). The syntax tree for a dictionary entry can be relatively long and a comparatively complex algorithm is required to extract the structure of the dictionary entry.

Yet once a coding sentence has been found to be syntactically correct only the structure of the sentence but not the sequence of productions which have been applied is of interest. Therefore a semantic tree is constructed by condensing the levels of a syntax tree to keep only the structurally relevant nodes. The semantic tree is comparable to an abstract syntax tree as described for example in Louden (1997). The term semantic tree is used here to emphasise the intention to represent the semantic content of the coding sentences.

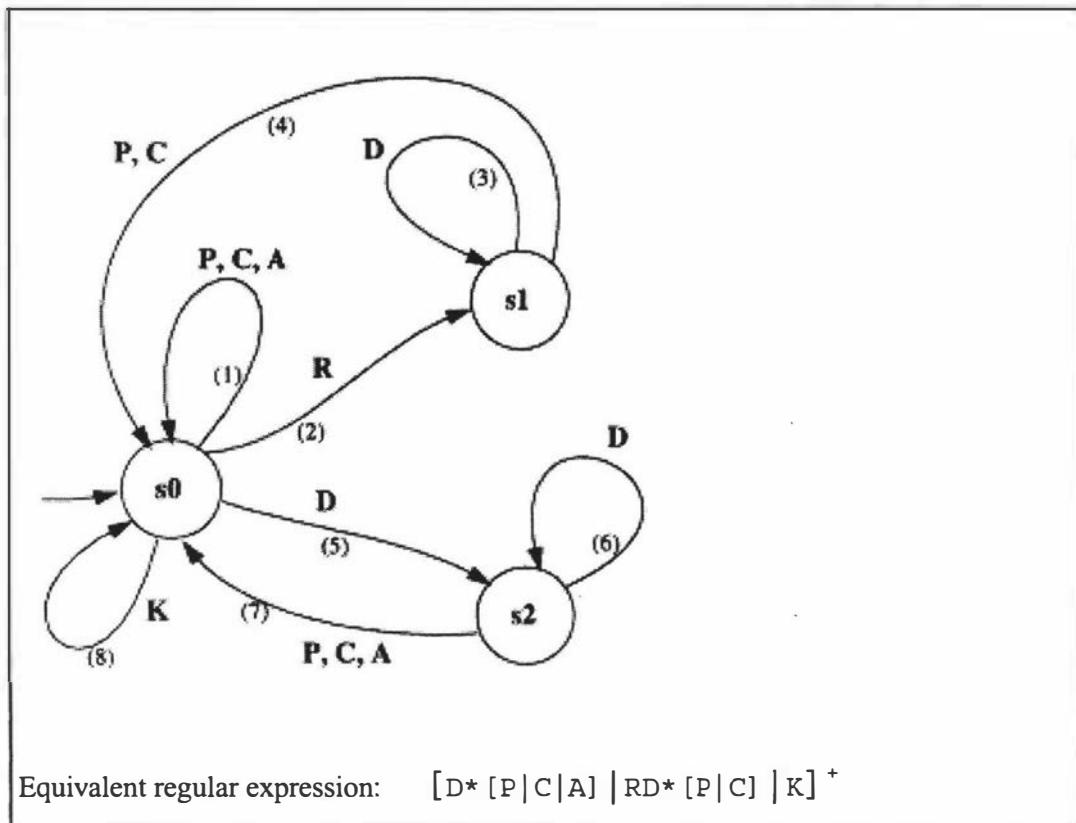


Figure 5.2: Automaton for first pass of syntax analysis with equivalent regular expression

The semantic tree for the FSCL grammar has a maximum level of four. The levels are the sentence level, the actor or concept level, the activity level and the object level. The clauses of a sentence become the branches of the semantic tree while the predicates of the sentence are parts of the ‘clause branches’. The advantage of storing a sentence in the form of a semantic tree is that the semantic roles as subject, verb or object of each sentence component are immediately visible. Further, by following the branches of the tree, the subject - verb - object relationships are clearly identified. With these properties the semantic trees support the querying of the coding sentences according to their meaning which is further outlined in Section 5.2.1.

The semantic trees for the coding sentences are constructed parallel to the syntax check and are stored in a relational database. Each semantic tree, representing a coding sentence, is stored in two parts. The first part is a textual pattern string giving the structure of the tree, the second part is the detailed information about each structural element of the tree.

Transition	Initial State	End State	Input	Output	Description
(1)	s0	s0	P or C or A	p or c or a	P or C or A without Descriptor or Preposition
(2)	s0	s1	R	r	Preposition introduces an object
(3)	s1	s1	D	null	Descriptors for the object
(4)	s1	s0	P or C	null	P or C which has been proceeded by Preposition
(5)	s0	s2	D	null	Descriptor which has to be followed by P or C or A
(6)	s2	s2	D	null	Additional Descriptor
(7)	s2	s0	P or C or A	p or c or a	P or C or A which has been proceeded by Descriptor
(8)	s0	s0	K	k	Conjunction

Table 5.1: Transition table for automaton

Coding sentence:	Bill hit Jim with ball and fast ran to teacher for help before end of interval.
Pattern string:	P A P R P K D A R P R C K C R C
Modified pattern string:	p a p r k a r r k c r

Figure 5.3: Coding sentence example with pattern string and modified pattern string

The pattern string is constructed by assigning a level number to each node or leaf of the semantic tree. The actor or concept level is assigned the number ‘1’, the activity level the number ‘2’, and the object level the number ‘3’. By going through the semantic tree in top-to-bottom, left-to-right direction the level numbers can be collected and stored as characters of a text string.

Associated with each node or leaf of the semantic tree, represented by a level number in the pattern string, is a database record which contains the detailed information about the words building the sentence.

By splitting the storage of the semantic trees for the coding sentences into the two parts of structural information and detail information the querying of the coding sentences is

Productions		Director Set
$\langle S \rangle$	$::=$	$\langle \text{sentence} \rangle \#$
$\langle \text{sentence} \rangle$	$::=$	p c
$\langle \text{restSentence} \rangle$	$::=$	$\#$ kp kc
$\langle \text{actorClause} \rangle$	$::=$	$p \langle \text{moreActors} \rangle \langle \text{groupedPredicate} \rangle$
$\langle \text{moreActors} \rangle$	$::=$	a p
$\langle \text{groupedPredicate} \rangle$	$::=$	$\langle \text{predicate} \rangle \langle \text{restGroupedPredicate} \rangle$
$\langle \text{restGroupedPredicate} \rangle$	$::=$	$kp, kc, \#$ ka
$\langle \text{predicate} \rangle$	$::=$	$a \langle \text{moreActions} \rangle \langle \text{object} \rangle$
$\langle \text{moreActions} \rangle$	$::=$	$r, p, c, k, \#$ a
$\langle \text{conceptClause} \rangle$	$::=$	$c \langle \text{object} \rangle$
$\langle \text{object} \rangle$	$::=$	$k, \#$ r p c

λ denotes the empty string

Figure 5.4: Generic LL2 grammar for FSCL

prepared (see Section 5.2.1). An example of a semantic tree and the associated pattern string is displayed in Figure 5.5.

5.1.5 Limitations of FSCL Compared to Natural Language

The following paragraphs summarise the limitations of the natural language like coding language FSCL compared to English as a natural language. The linguistic terms used are drawn from Wardhaugh (1995).

- FSCL does not support causal variations like passives, imperatives, questions, and negatives as recognised sentence structures. There are ways, however, to express negation within FSCL sentences. The user can define a Descriptor ‘not’ for use in a sentence like ‘Peter not plays’. The absence of something (as in ‘no noise’) can be expressed by the definition of a suitable word (like ‘silence’ to describe the fact that no

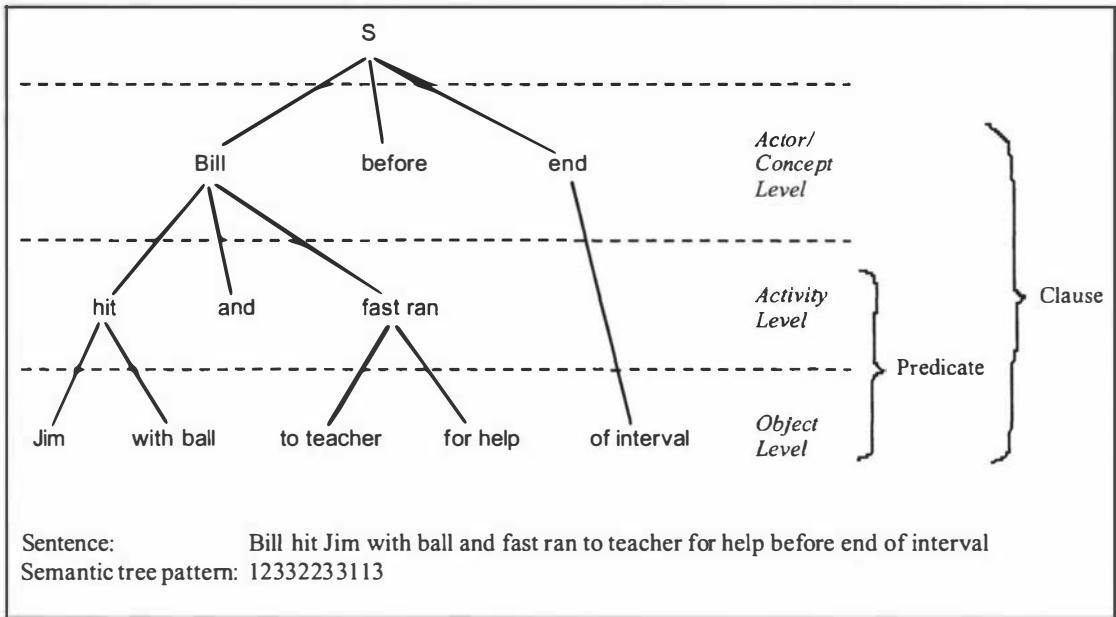


Figure 5.5: Semantic tree for example sentence

noise can be heard). This second approach can be supported by arranging the words in word hierarchies to support convenient retrieval. Examples of these approaches to express negation are given in the descriptions of the case studies (see Chapter 7).

- FSCL does not support the use of pronouns. While pronouns like 'he', 'she', 'they' can be introduced as words in the category Person/Thing FSCL has no facility to associate a pronoun with the respective person by evaluating the context of the sentence or clause. The user of FSCL has to circumvent this deficiency of FSCL by explicitly stating the person or thing referred to.
- To make it possible to analyse the coding sentences with a relatively simple grammar, FSCL restricts the use of conjunctions to the combination of clauses and verb phrases. In the following cases, where natural language would use conjunctions, FSCL suppresses the use of conjunctions:
 - Where multiple actors perform exactly the same activities on the same objects the actors are not joined with a conjunction but an 'and' is assumed;
 - Where an actor performs multiple activities on the same objects the activities are not joined with a conjunction but an 'and' is assumed;
 - Where an activity relates to several objects the objects are not joined with a conjunction but an 'and' is assumed.

Example sentences are:

Natural language: Peter **and** Paul play on the swing.

FSCL:	Peter Paul play on swing.
Natural language:	Peter runs and jumps.
FSCL:	Peter runs jumps.
Natural language:	Paul plays on the slide and the swing.
FSCL:	Paul plays on slide on swing.

Where conjunctions can be used in FSCL, in the combination of clauses and verb phrases, FSCL does distinguish between the different conjunctions. The meaning in FSCL sentences like ‘Peter cries **before** talking to teacher’ and ‘Peter cries **after** talking to teacher’ is preserved. Additionally, the query language FSQL provides Boolean query options where ‘AND’ and ‘OR’ are treated as expected.

- FSCL does not support constructs which use auxiliary or modal verbs. The user can define auxiliary or modal verbs in the category Activities and can use them in combination with other verbs yet FSCL will not recognise the special grammatical function of these verbs. In sentences like ‘Peter is going on slide’ or ‘Peter can go on slide’ FSCL will interpret that the actor ‘Peter’ is performing two parallel activities, ‘is’ and ‘going’ in the first example, ‘can’ and ‘go’ in the second example, which both refer to the object ‘on slide’. The user can formulate sentences like these to produce more natural language like constructs. With the query language FSQL (see Section 5.2), sentences containing the information that ‘Peter is going somewhere’ can be extracted based on the different forms of the verb ‘to go’, by ignoring the auxiliary and modal verbs if desired.
- Phrasal verbs have to be defined in FSCL as one word which leads to unusual constructs for transitive phrasal verbs.

Natural language:	Peter turns the light out .
FSCL:	Peter turns out light.

The phrasal verb ‘turns out’ is defined in FSCL as **one** word as words in FSCL can consist of string combinations containing the blank character. Semantically, ‘turns out’, which could as well be defined as ‘turns-out’, is one word in FSCL.

- FSCL makes no distinction between adjectives and adverbs on a category basis. While it is up to the user to define both forms of a descriptive word, like ‘slow’ and ‘slowly’, the descriptive word is always positioned before the word it refers to. While this approach facilitates the correct association between descriptor and word described, it introduces, in the case of adverbs, an incorrect word order.

Natural language:	Peter runs slowly .
FSCL:	Peter slowly runs .

The definition of two separate categories for adjectives and adverbs would have allowed the positioning of adverbs correctly following the verb. The decision to combine adjectives and adverbs has been made to simplify the development of the vocabulary. In

cases where the word forms of adjectives and adverbs are very similar ('slow' and 'slowly') the user might choose to define only one of the words. While the sentences 'Peter slowly runs' or 'Peter slow runs' are not elegant they are still fully understandable to a human reader.

- FSCL does not support inflections of words. Again, the user has the ability of defining words in their different inflections and use the appropriate form accordingly. The use of word hierarchies can support the later querying of all forms with one query.

Category Activities: play

plays
is playing
are playing
played

FSCL: Peter plays.

Peter is playing.

Peter Paul play.

Peter Paul are playing.

Peter Paul played.

Using the word hierarchies all these behaviour descriptions could be retrieved by the one query 'Peter play' (see Section 5.2.2 on flexibility in querying for further details).

5.1.6 Coding Procedure

The first step in using FSCL is to set up the vocabulary. Every word which is to be used for the description of the behaviours has to be defined. The words have to be assigned to a category, words can be arranged in hierarchies and word links can be specified. In the current implementation of FSCL within the program PAC (see Chapter 6), the vocabulary is defined on a domain level and can be used in a number of studies of that domain. The vocabulary can be extended and modified during the course of the study.

After the vocabulary is defined, the coding of the behaviour displayed on video (or on other media types) can start. Coding sentences are formulated and linked to particular sequences of the video file. These coding sentences, the descriptions of the behaviour displayed on the video, are later used for locating specific behaviours on the video. To be able to do this, the semantic content of the coding sentences, expressed in the sentence structures, has to be accessible. To ensure the correctness of the sentence structures all coding sentences are syntax checked before they are accepted as valid descriptions. Valid FSCL coding sentences are converted into a semantic tree format and stored as such in a database. The semantic tree format is illustrated in Section 5.1.4.2 on the formal specification of FSCL. Coding sentences which do not conform to the syntax of FSCL can be either discarded or

corrected by the user. The term *code instance* is used to refer to the combination of a coding sentence and the attached behaviour recording segment.

5.2 The Flexible Structured Query Language (FSQL)

The Flexible Structured Query Language FSQL is based on the Flexible Structured Coding Language FSCL. The purpose of FSQL is to extract code instances from a database of code instances according to common characteristics. These characteristics can be based on the coding sentences of the code instances, on the behaviour recording segments attached to the code instances, or on combinations of code instances.

A basic query can be formulated according to the syntax rules for a coding sentence. The evaluation of a basic query retrieves a selection from a set of code instances which correctly and completely represents the meaning of the query in the relevant coding sentences (see Section 5.2.1). A number of parameters can be set to further qualify the basic queries (see Section 5.2.2). More complex query statements, constructed around one or two basic queries, evaluate combinations of code instances based on Boolean, position or sequence properties of single code instances or groups of code instances (see Section 5.2.3).

5.2.1 Requirements of Correctness and Completeness

Section 4.3.1 outlines the problems of using a text search mechanism to retrieve description sentences without considering the grammar implicit to natural language. With a pure text search strategy it is impossible to retrieve only semantically correct description sentences and, at the same time, retrieve all semantically matching description sentences. FSCL and FSQL have been designed to allow the correct and complete retrieval of coding sentences by basing the query evaluation on the grammar structure of coding and query sentences. Using the two example sentences from Section 4.3.1, ‘Bill hit Jim and ran’ and ‘Bill hit Jim and Jim ran’, the query evaluation will be illustrated.

The two example sentences contain words of the vocabulary which would have been defined as belonging to the categories as presented in Table 5.2. Based on the categories, the words of the sentences belong to, and on the generic grammar, defined on these categories, the two sentences are syntactically correct and can be presented in form of semantic trees (see Figure 5.6), as has been defined in Section 5.1.4.2.

As in the example in Section 4.3.1, the query ‘Bill ran’ will now be applied to the two example coding sentences. This query is, in terms of FSQL, a ‘basic’ query as it is fully based on the properties built into FSCL. The query, consisting of two words of the categories Person/Thing and Activity, is parsed according to the syntax rules of FSCL. The

Category	Word
Person/Thing	Bill Jim
Activity	hit ran
Conjunction	and

Table 5.2: Vocabulary of two example sentences

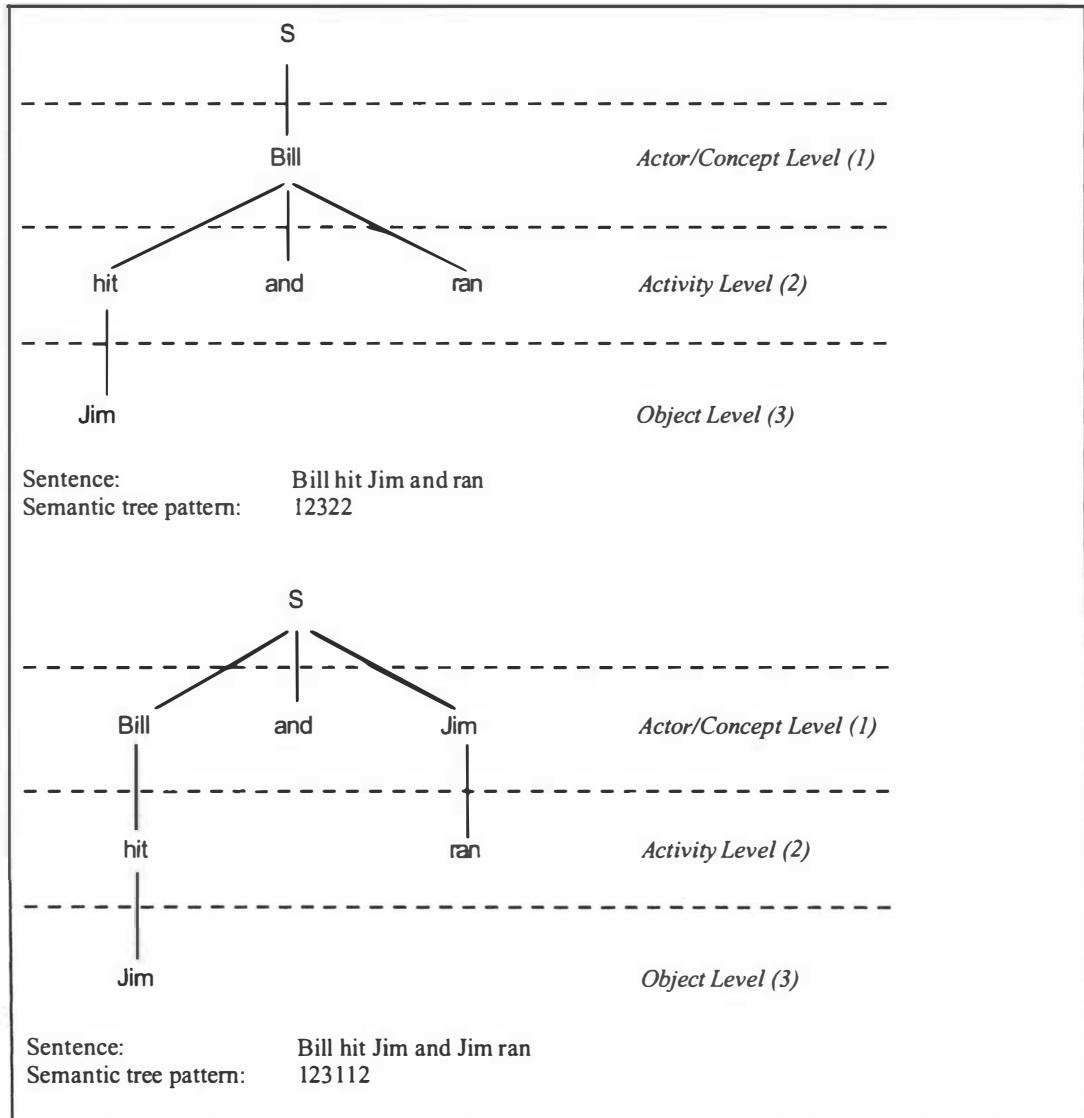


Figure 5.6: Semantic trees for two example sentences

semantic tree for the query sentence, called the ‘basic query tree’, is constructed (see Figure 5.7).

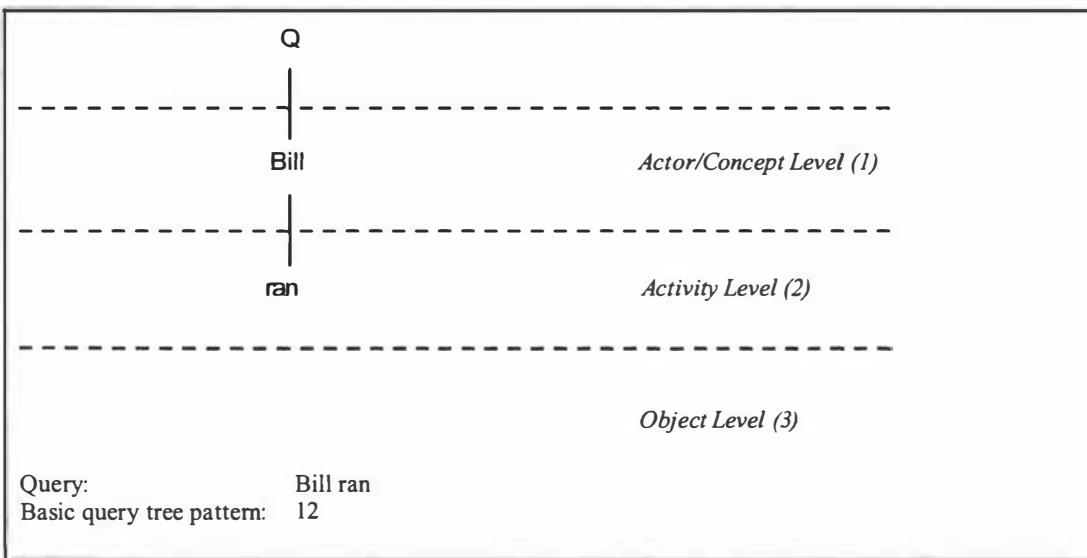


Figure 5.7: Query tree for example query

To evaluate which of the coding sentences fulfils the query, the semantic trees and the basic query tree have to be compared. Looking at the graphical representation of the trees this can be easily done by looking for matching tree branches. The first semantic tree contains the 'Bill ran' branch of the basic query tree and therefore identifies a coding sentence which contains the behaviour description asked for in the query. The second semantic tree does not contain the 'Bill ran' branch of the basic query tree and therefore does not identify a coding sentence containing a correct answer to the query.

This example shows how the grammar approach taken in FSCL and FSQL facilitates the correct and complete evaluation of the query. Because each word of the vocabulary belongs to one of the categories on which the grammar rules are defined it is possible to identify the role of each word in the sentence and the relationships of the words in the sentence to each other. Once these relationships are identified it is obvious that in the second example sentence no relationship exists between the words 'Bill' and 'ran' and therefore this sentence cannot represent a correct answer to the query asked. This identification of the relationships of the words in the sentences facilitates the **correct** evaluation of queries.

The **complete** evaluation of queries is also linked to the presentation of relationships between the words. In the first example sentence the words 'Bill' and 'ran' are physically separated by the words 'hit Jim and' yet logically the actor 'Bill' and the activity 'ran' belong together. This logical connection is discovered by analysing the structure of the sentence and is expressed in the semantic tree.

In an implementation of FSQL the query evaluation of finding matching tree branches can be defined via the semantic and basic query tree patterns. These patterns represent the structure of the sentences and allow tree branches to be found which have the correct form on a category basis. Once matching branches on a category level are identified it can be checked if the individual words match and therefore a coding sentence fulfils the query. The details of such a query evaluation algorithm are explained in Section 5.2.6 and a justification for this two step approach is given in the same section.

5.2.2 Flexibility in Querying

The previous section has shown how basic queries are evaluated according to the structure of the coding sentences. In this section the elements of FSQL will be introduced which create the flexibility in using the query language. These elements are listed here and explained in detail in the following paragraphs:

- Suppression of detail in a query;
- Use of wild cards;
- Use of word relationships on a vocabulary basis.

Suppression of Detail in a Query

FSCL has been designed to facilitate a rich description of behaviour containing prepositions and descriptors, allowing multiple actors, activities and objects. When formulating a query, the user can decide the level of detail incorporated into the query. If the user omits prepositions and descriptors in the query, the query will be evaluated by disregarding prepositions and descriptors contained in the coding sentences. If on the other hand a preposition or descriptor is given in the query then it has to be matched in the coding sentence.

One activity, or a concept in a ‘concept clause’ (see Section 5.1.4), can have several objects attached to it. In a query these objects can be omitted. Yet if the objects are specified in the query, they have to be given in the correct sequence.

FSCL allows the specification of parallel actors, performing exactly the same sequence of related activities, or the specification of parallel activities, relating to the exactly the same sequence of objects (see Section 5.1.4). The user can decide to query for only one of the parallel actors or activities. Yet if two of the parallel actors or activities are specified they have to be given in the correct sequence.

Use of Wild Cards

FSQL defines four wild card operators:

ANY-PERSON/THING

ANY-ACTIVITY

ANY-CONCEPT

ANY-CONJUNCTION

These operators can be placed in a query in place of a word of the respective categories. These four wild card operators have been defined because words of the categories Person/Thing, Activity, Concept and Conjunction are necessary to form the structure of a sentence. Wild card operators are not necessary for the categories Preposition and Descriptor because words of these categories can be omitted from a query as described above.

Use of Word Relationships on a Vocabulary Basis

Relationships between words in the form of hierarchy and word link relationships can be built into the vocabulary in the preparation of a study (see Section 5.1.3) or can be defined as concept relationships in the interpretation phase of a study (see Section 4.4). By setting parameters before executing a query the user can choose to make use of these relationships for the evaluation of the query.

Setting the ‘Use Hierarchies’ parameter has the effect that a query is evaluated considering not only the words in the query but as well all words which are hierarchically ‘below’ the query words. This feature allows the formulation of queries on a generic, conceptual basis and makes querying for related words convenient. Setting the ‘Use WordLinks’ parameter causes the query language to substitute words in the query with all words which are linked to the query words.

While hierarchies and word links are defined on a domain basis within the vocabulary, the concept relationships are defined on a study basis. Setting the ‘Use ConceptRelationships’ parameter causes words in the query, which in this case have to be words of the category Concepts, to be substituted with words which have a concept relationship to the query words.

Using the example vocabulary defined in Figure 5.1 the coding sentences and queries given in Table 5.3 illustrate the flexible query options.

5.2.3 Combination of Code Instances through Querying

While the two previous sections have dealt with the query evaluation within one coding sentence, arising directly from the specification of FSCL and termed ‘basic’ queries, the following paragraphs introduce the evaluation of combinations of code instances through *Boolean*, *time/position*, *sequence* and *combined-with* query operators. Two general

Coding Sentence	Query Sentence	Query Evaluation
Bill strong hit Jim	Bill hit Jim	Suppression of Descriptors
Bill hit Jim	Bill hit	Suppression of Object
Bill hit	ANY-PERSON/THING hit	Wild card for Actor
Bill hit	Bill ANY-ACTIVITY	Wild card for Activity
Bill hit Jim	Senior Students hit Junior Students	Use of Hierarchies
Jim Clare jump on slide	Jim jump on slide	Suppression of parallel Actor
Clare jump	Female Students jump	Use of WordLinks
happiness Bill	joy Bill	Use of concept relationship as could be defined between 'happiness' and 'joy' during the course of the study

Table 5.3: Flexible query options

remarks are important for the understanding of the combination of code instances in the context of querying:

- Every query is applied to a set of code instances (for the definition of the sets see Section 6.4). The result of the evaluation of a basic query is a subset of this set of code instances. Combination query operations can be performed within a subset or between subsets.
- A code instance consists of two parts, the coding sentence and the attached behaviour recording segment. The basic query operations are applied to the coding sentences. Further query options introduced in the following paragraphs are applied to the behaviour recording segments.

The query options introduced in this section can be combined within one query. Boolean queries are applied to the results of basic queries or other Boolean queries. Time/Position, sequence and combined-with queries are applied to the results of Boolean queries or basic queries, when no Boolean query is specified. The exact range of possible query combinations can be taken from the formal specification of FSQL in Section 5.2.5. Figure 5.8 shows the different layers of a query and indicates which elements of the code instances in a set are examined in each part of a query.

Time/Position Sequence Combined-With	Queries	examining	Combinations of Boolean Query Results based on Time/Position Properties of Behaviour Recording Segments
Boolean OR AND NOT	Queries	examining	Boolean Combinations of Basic Query Results
Basic Queries		examining	Content of Coding Sentences based on FSCL

Figure 5.8: Query layers

Boolean Query Options

FSQL contains the Boolean operators OR, AND, and NOT which are applied to the results of basic queries or other Boolean queries. The result of a Boolean query evaluation is a set of code instances. Figures 5.9, 5.10 and 5.11 show examples of Boolean query evaluations.

Time/Position Query Options

The time/position query options of FSQL are applied to the results of the Boolean queries and target the behaviour recording segments attached to the code instances. For time based media like video and audio recordings time values are compared, for text documents or images positions are compared.

FSQL contains the binary time/position comparison operators BEFORE, SAME-AS, and AFTER and the time/position interval operators FROM and TO. Additionally, a constant expression named ‘ALL_DESCS’ is defined. With these operators queries of the following types can be formulated:

- BooleanQueryResult BEFORE/SAME-AS/AFTER BooleanQueryResult WITHIN time/positionValue;
- BooleanQueryResult FROM time/positionValue TO time/positionValue;
- ALL_DESCS FROM time/positionValue TO time/positionValue.

For the evaluation of time/position query options a number of aspects arising from the nature of the code instances have to be taken into consideration:

- The behaviour recording segments attached to the code instances can refer to different study files of different media types. Time/Position comparisons can only be applied to

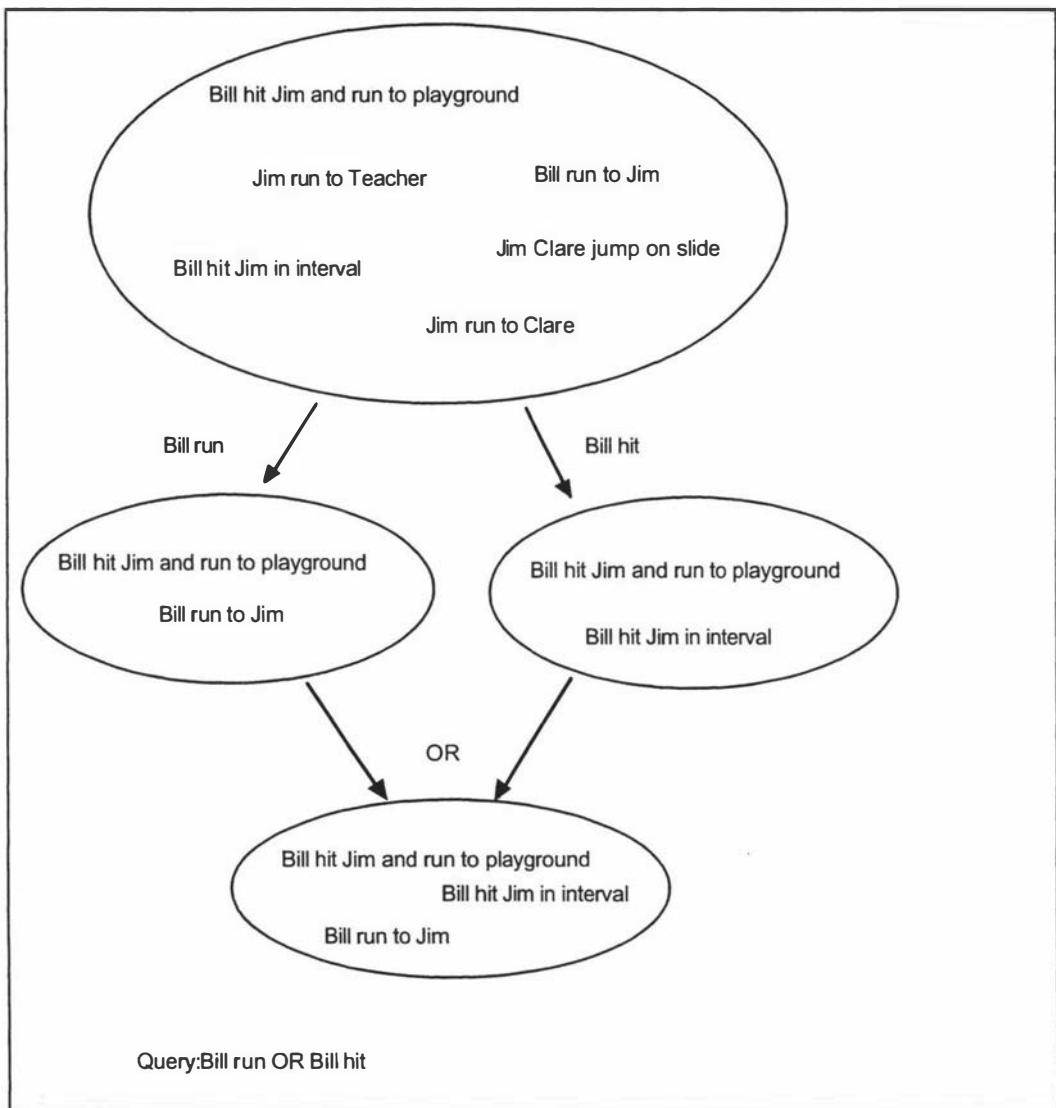


Figure 5.9: Boolean OR query

two Boolean results for which the behaviour recordings refer to the same study file or to study files which share a common time line (as described in Section 4.1).

- With a multimedia computer system time intervals are usually captured at a high resolution which might be more accurate than intended by the analyst. The segment boundaries for a video based behaviour recording for example could be set to a frame level accuracy while the analyst might only be concerned with a one second resolution. This can have an effect on time based comparisons when the analyst regards two events happening at the ‘same’ time if they were only half a second apart while the computer system would accurately determine that the two events were several frames apart and therefore not happening at the same time. To allow the analyst to determine

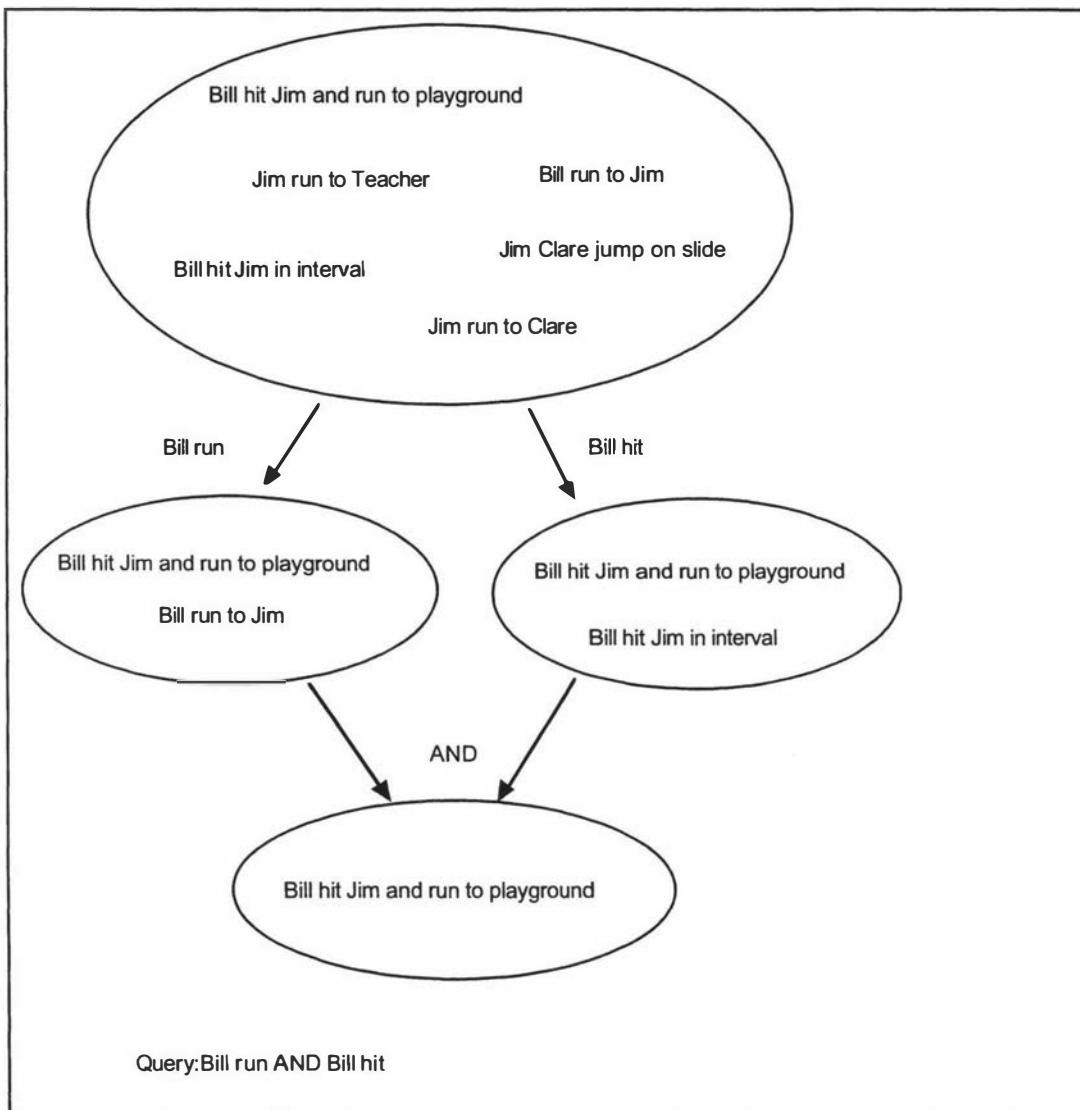


Figure 5.10: Boolean AND query

the required level of resolution, on an individual query basis, FSQL supports the specification of a time frame value with the ‘WITHIN time/positionValue’ query option. The effect of this specification can be seen in Figure 5.12.

- Using the comparisons BEFORE and AFTER the analyst aims to extract two behavioural events which occurred in the specified sequence. Yet these sequences of events might only be of interest to the analyst if they both occurred in relative close proximity to each other. Looking at a causal relationship of two events, a ‘kick to goal’, for example, might only be relevant to a subsequent ‘goal’ if the two events occur within less than one second. To determine the required proximity of two events on an

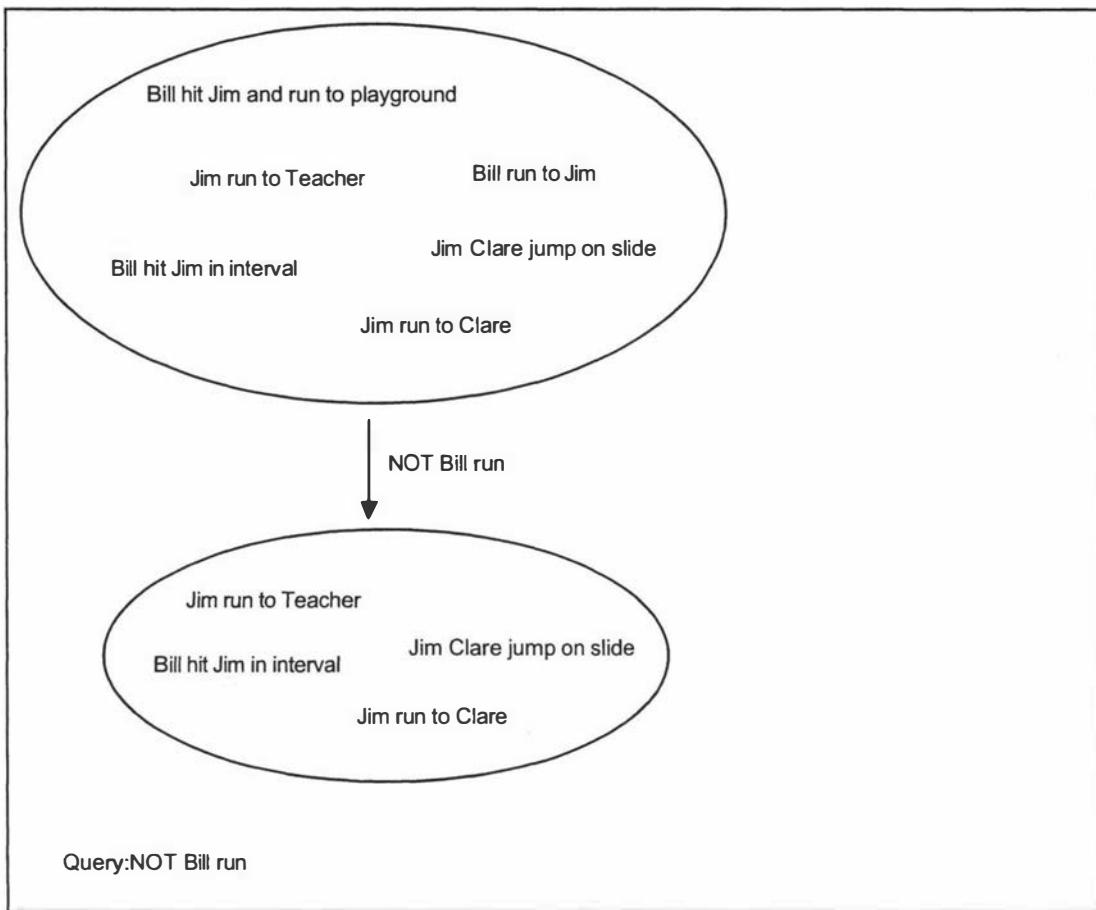


Figure 5.11: Boolean NOT query

individual query option the analyst can again use the ‘WITHIN time/position value’ query option with its effect being displayed in Figure 5.12.

- The behaviour recording segments can refer to points-of-time or periods-of-time (or single characters or a sequence of characters respectively). Point-of-time intervals (or single character intervals) are treated as special cases of periods (or sequences), having the same values for start and end. Different interpretations are possible in comparing two time periods (or analogue sequences of characters) regarding the operators BEFORE/SAME-AS/AFTER (Roddick and Patrick, 1992; Wahl and Rothermel, 1994). In this work the following interpretations have been chosen:
 - BEFORE: Period1 is regarded as BEFORE period 2 if the start of period 1 occurs before the start of period 2.
 - SAME-AS: Period 1 is regarded as SAME-AS period 2 if the two periods have some overlap.

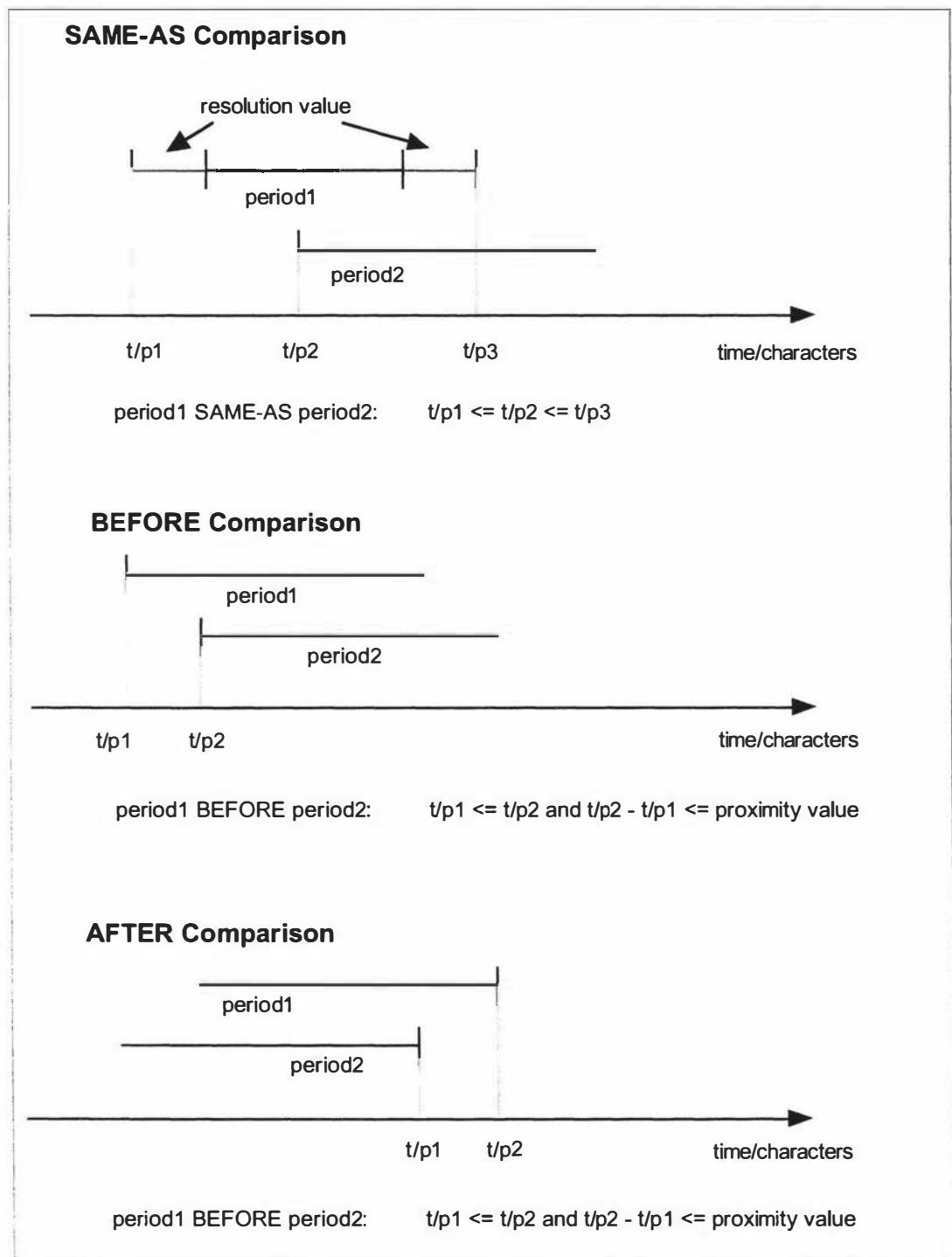


Figure 5.12: Time/Position comparison evaluation

- AFTER: Period 1 is regarded as AFTER period 2 if the end of period 1 occurs after the end of period 2.

A graphical display of the evaluation of the time/position comparisons including the specification of resolution or proximity is given in Figure 5.12.

Sequence Query Option

The sequence query option FOLLOWED-BY is provided to support the ‘pairing’ of events in the style of ‘start’ and ‘end’ events. This query option does not consider the distance in time or characters between two events but the sequence of events selected through the preceding Boolean queries. The sequence query is evaluated as

BooleanQueryResult FOLLOWED-BY BooleanQueryResult

by pairing an event belonging to the result of the first Boolean query with an event of the second Boolean query without any other results of either Boolean query being ‘between’ these events. Like the time/position queries this query can only be applied to code instances referring to the same study file. Figure 5.13 illustrates the evaluation of this query option.

Combined-with Query Option

The combined-with query option provides the ability to query across related but different behaviour recordings. The time/position and sequence based query options introduced so far can only compare code instances which refer to the same behaviour recording. The combined-with query option allows linking of code instances relating to two different behaviour recordings by accessing specific code instances defined to provide the connection between the two behaviour recordings. A typical situation for using this query option would be if a sequence of behavioural events had been recorded continuously on a time based medium like video and that some form of commentary had been applied to this sequence of behavioural events. This commentary would refer to the sequence of events having its own commentary time line (or no time line at all if transcribed), possibly not in continuous order, possibly referring selectively to single events in the video recordings.

The following steps are necessary to prepare and perform the combined-with querying:

- The behaviour recording showing the continuous sequence of events is identified as the ‘primary’ behaviour recording. This primary behaviour recording is divided into consecutive, non-overlapping segments by applying distinct codes as coding sentences.
- In the commentary behaviour recording, the ‘secondary’ behaviour recording, sequences are identified which refer to the segments of the primary behaviour recording. These sequences are coded with the same segment identifiers as used for the primary behaviour recording. The segmentation applied to the second behaviour recording can be repetitive and overlapping.

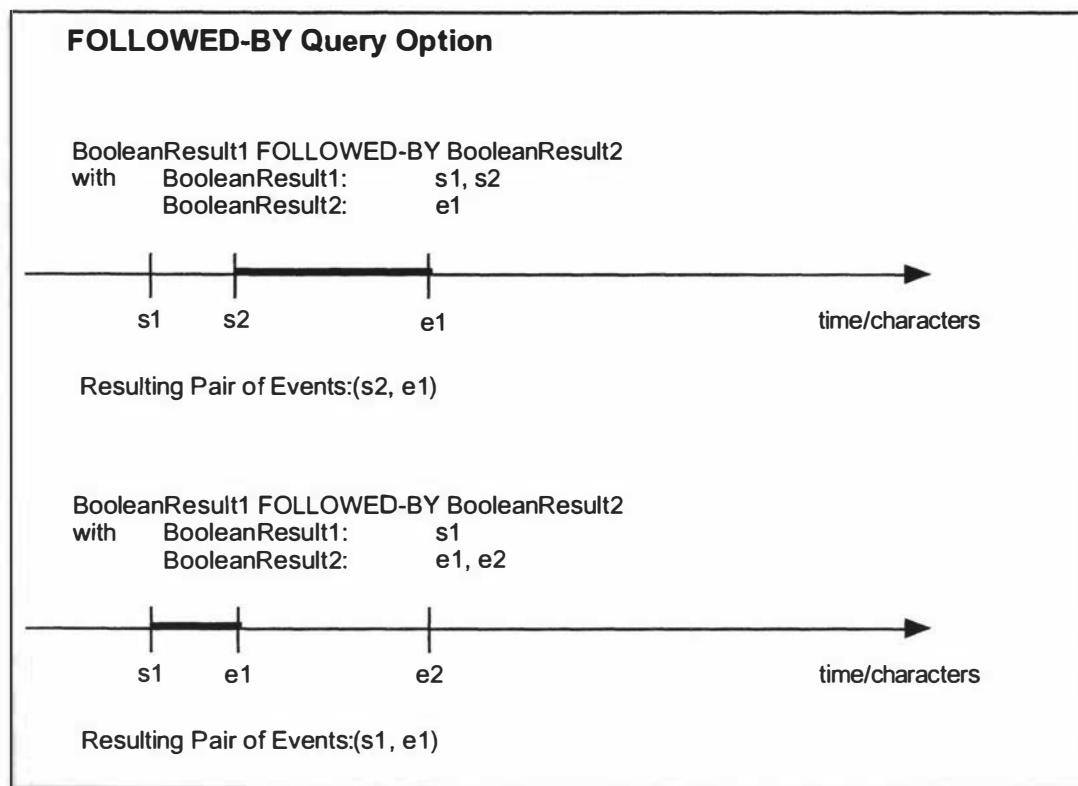


Figure 5.13: Sequence query option

- The coding of behaviour for both behaviour recordings is performed as usual.

The combined-with query option can be applied by specifying the segment codes as connecting elements between the two behaviour recordings:

- BooleanQueryResult COMBINED-WITH BooleanQueryResult VIA segment code
- The segment code can be either a single code or the identifier of a hierarchy of codes as defined in the vocabulary of FSCL. In the latter case the combination of Boolean query results is attempted for each of the codes of the hierarchy.

A short illustration of the combined-with query option is shown in Figure 5.14, a fuller, more detailed example is given in the ‘Reading Groups’ case study in Section 7.2.1.

5.2.4 Format of the Query Results

The result of the final query evaluation, regardless of the complexity of the query, is always independent from the specification of both the grammar and the vocabulary of FSCL. The query result consists of several query result instances. Each query result instance has two parts, a text string derived from one or two coding sentences and one or

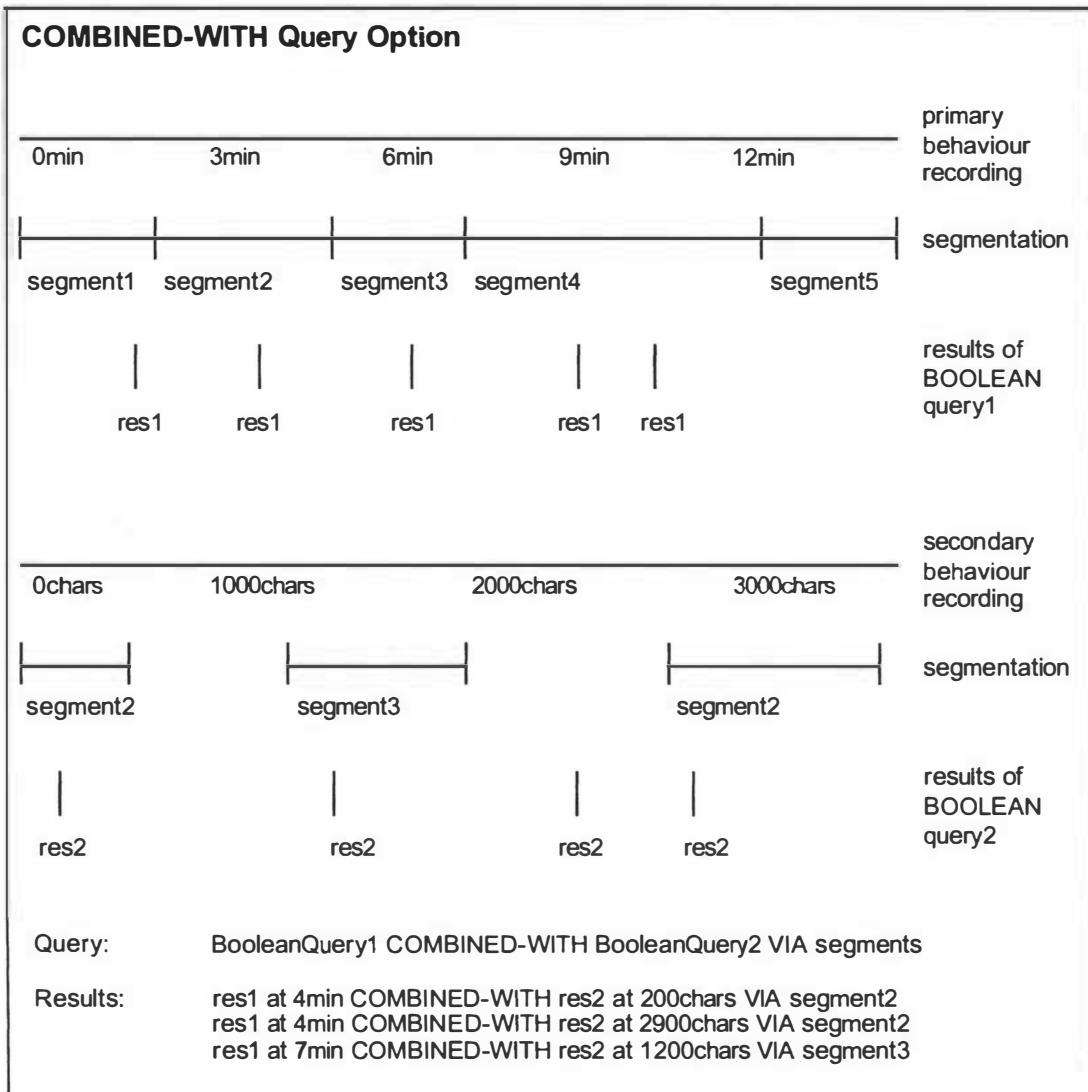


Figure 5.14: Combined-with query option

two behaviour recording segments. In the course of the query evaluation the coding sentence, consisting of separate words, is converted into a single text string.

Basic Query Results

The query result of a basic query is formed by a set of query result instances. Each query result instance is based on exactly one code instance. The coding sentence of the code instance is converted into a single text string. This is done by concatenating the words of the coding sentence to one text string. The text string of the query result instance is therefore independent from the vocabulary. The behaviour recording segment of the code instance stays unmodified.

Boolean Query Results

The Result of a Boolean query consists of a selection of query result instances sourced from the basic query results the Boolean query is built on. The individual query result instances are not modified.

Results of Time/Position Queries Based on One Boolean Query

These results are selections of query result instances from the Boolean query results. The individual query result instances are not modified.

Results of Time/Position Queries Based on Two Boolean Queries and of Sequence Queries

These results consist of query result instances which are constructed from one query result instance of each of the two Boolean query results. The two behaviour recording segments, both referring to the same study file, are connected to one larger segment by taking the smaller of the start values and the larger of the end values as new segment boundaries. The text string of the query result is constructed by concatenating the two individual text strings and inserting the relevant query operator for information:

```
queryResultText1 BEFORE queryResultText2  
queryResultText1 SAME-AS queryResultText2  
queryResultText1 AFTER queryResultText2  
queryResultText1 FOLLOWED-BY queryResultText2
```

Combined-With Query Results

The query result instances forming these results are again constructed based on two query result instances originating from two Boolean query results. In comparison to time/position and sequence queries, these results have to retain a reference to two separate behaviour recording segments as these segments refer to different study files which cannot be combined into one segment. The text strings contain the texts of the two Boolean query results and the information via which segment code the connection between the two instances has been established:

```
queryResultText1 COMBINED-WITH queryResultText2 VIA segmentCode
```

Appendix A gives a summary of all query options provided by FSQL.

5.2.5 Formal Description of FSQL

FSQL is defined as an LL2 grammar and builds on the specification of FSCL (see Herschel, 1974, or Louden, 1997, for a definition of LL2 grammars). Figure 5.15 shows the grammar specification for FSQL.

Productions	Director Set
$\langle Q \rangle ::= \langle query \rangle \#$	
$\langle query \rangle ::= \langle booleanQuery \rangle \langle restQuery \rangle$ ALL_DESCS FROM TIME TO TIME	p, c, d, NOT ALL_DESCS
$\langle restQuery \rangle ::= \lambda$ FOLLOWED-BY $\langle booleanQuery \rangle$ COMBINED-WITH $\langle booleanQuery \rangle$ VIA $\langle word \rangle$ $\langle timeOperator \rangle \langle booleanQuery \rangle$ WITHIN TIME FROM TIME TO TIME	# FOLLOWED-BY COMBINED-WITH BEFORE, SAME-AS, AFTER FROM
$\langle timeOperator \rangle ::=$ BEFORE SAME-AS AFTER	
$\langle booleanQuery \rangle ::= \langle orLevel \rangle \langle restOrLevel \rangle$	
$\langle restOrLevel \rangle ::= \lambda$ OR $\langle orLevel \rangle \langle restOrLevel \rangle$	#, FOLLOWED-BY, BEFORE, SAME-AS, AFTER, FROM OR
$\langle orLevel \rangle ::= \langle andLevel \rangle \langle restAndLevel \rangle$	
$\langle restAndLevel \rangle ::= \lambda$ AND $\langle andLevel \rangle \langle restAndLevel \rangle$	#, FOLLOWED-BY, BEFORE, SAME-AS, AFTER, FROM AND
$\langle andLevel \rangle ::= \langle sentence \rangle$ NOT $\langle sentence \rangle$	p, c, d NOT
$\langle word \rangle ::= P$ A C	P A C
$\langle sentence \rangle ::=$ (see specification of grammar for FSCL)	
λ denotes the empty string	

Figure 5.15: LL2 grammar for FSCL

5.2.6 Query Evaluation Using FSCL

The evaluation of a query has two main phases. In the first phase the query is syntax checked, the query parameters are evaluated and the full query tree is constructed. In the second phase the query, utilising the full query tree, is applied to the set of code instances as defined before submitting the query (see Section 6.4 on the definition of sets).

5.2.6.1 Construction of Full Query Tree

Section 5.2.1 described the construction of basic query trees for the basic queries, where basic queries are the parts of a query which fully refer to the structures of FSCL. The full query tree contains all elements of a query, that is the query options based on the basic queries (as described in Section 5.2.1) and the basic queries themselves.

The nodes and leaves of the full query tree are objects which describe the types of query to be performed and contain links to the ‘lower’ nodes or leaves identifying the subqueries which can be Boolean or basic queries. The nodes of a full query tree represent either time/position, sequence, combined-with or Boolean queries. The leaves of a full query tree have to be basic query trees because all other query types are built on basic queries.

A full query tree is constructed by parsing a query using both the grammars of FSQL and FSCL. The exact depth of a query tree cannot be predetermined. The Boolean query options are defined in a recursive way and handled in the full query trees as sequence of binary operations. The full query trees can be divided into three levels with only the Boolean level being of variable depth:

1. Level: Time/Position, sequence, combined-with;
2. Level: Boolean;
3. Level: Basic query.

An example of a full query tree indicating the query operations is given in Figure 5.16.

5.2.6.2 Application of Full Query Tree to Set of Code Instances

Every query is applied to a set of code instances (for the definition of sets see Section 6.4). The code instances conform to the FSCL grammar and are presented in the form of semantic trees (see Section 5.1.4.2).

To identify the code instances which fulfil the query conditions, the query evaluation algorithm works through the full query tree in top-to-bottom, left-to-right order. If the algorithm reaches a leaf, the basic query represented by the leaf is evaluated and the query results are passed on to the higher node of the tree. After both sub-branches of a tree are evaluated the query represented by the node is evaluated and the results are passed on to the higher level node. This process continues until the query is fully evaluated.

The evaluation of queries represented by nodes of the full query tree, the time/position, sequence, combined-with and Boolean queries, has been outlined in Section 5.2.3. The evaluation of full query tree leaves, representing the basic queries, has been briefly described in Section 5.2.1 and will be explained in more detail in the later paragraphs of this section. As part of the evaluation of the full query tree the code instances identified are

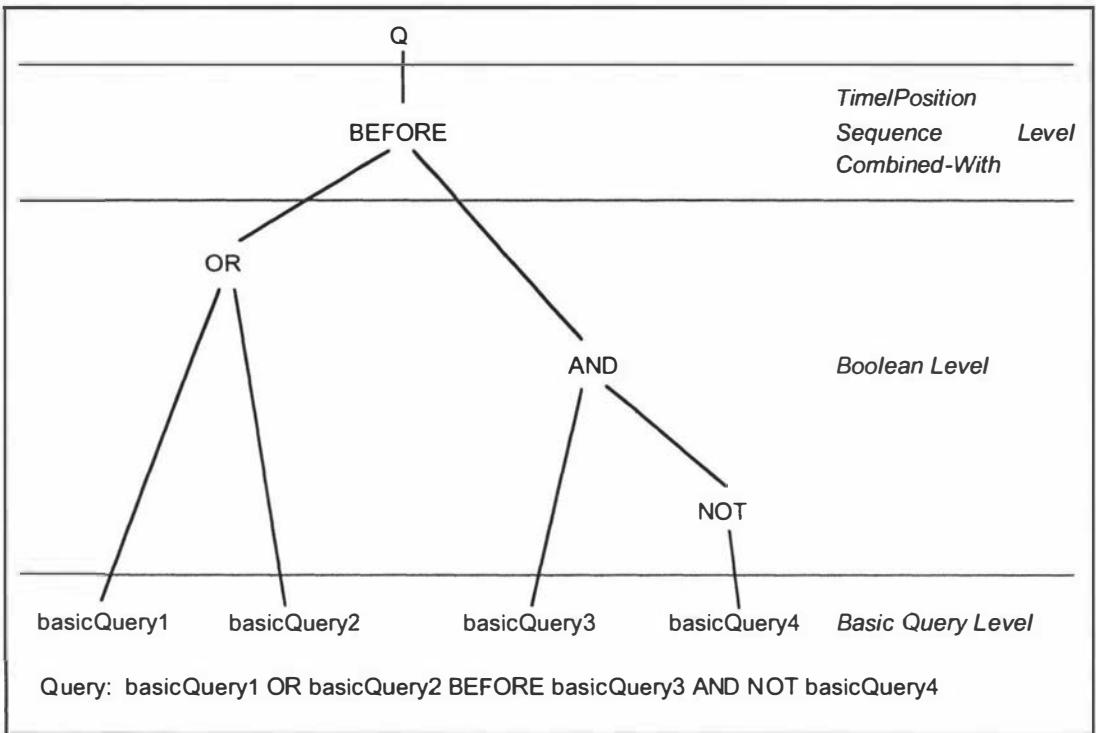


Figure 5.16: Full query tree example

converted into query result instances and where necessary query result instances are combined according to the specification in Section 5.2.4.

Match of Basic Query Trees and Semantic Trees

The evaluation of the basic queries stored in the leaves of the full query tree requires a matching between the basic query trees and the semantic trees (see Section 5.2.1). Both the basic query trees and the semantic trees are represented as semantic tree patterns indicating the sequence of nodes and leaves according to their level in the trees (see Section 5.2.1). The matching algorithm has two elements which are interwoven:

- The comparison of the basic query tree pattern and the semantic tree pattern to identify parts of coding sentences which have the same sentence structure as the query;
- The comparison on a word level to test for word matches within parts of coding sentences which have the correct sentence structure; this comparison has to consider the query parameters set for the query.

The comparison of the tree patterns is based on the following characteristics of these patterns:

- The patterns indicate the level of an element of a sentence:
 - 1: Actor/Concept Level;

- 2: Activity Level;
 - 3: Object Level.
- The tree structure is represented in the patterns:
 - An activity represented by a ‘2’ has been performed by the actor represented by a ‘1’ closest to the ‘2’ in left direction;
 - An object represented by a ‘3’ belongs either to an activity represented by a ‘2’ closest to the ‘3’ in left direction or to a concept represented by a ‘1’ closest to the ‘3’ in left direction.
- The comparison on a word level has to occur in the context of the query parameters and the definitions in the vocabulary (see Section 5.2.2):
- The word match might have to be done not just for the exact word in the query but for:
 - every word which is defined in the vocabulary hierarchically below the query word;
 - every word which is linked to the query word via a word link;
 - every word which has been defined as equivalent in the concept relationships.
 - Where wild cards are used in the query the word match can be performed on a category level.

The matching algorithm is presented in Figure 5.17 formulated in pseudo code. Further illustration of the matching algorithm using a number of example queries is given in Appendix B.

5.3 Conclusion

In this chapter the Flexible Structured Coding Language, FSCL, and the Flexible Structured Query Language, FSQL, were introduced. FSCL allows formulating natural language like sentences for the description of behaviour. The resulting coding sentences express subject – verb – object relationships and facilitate a detailed, rich description of behaviour including the use of adjectives, adverbs, conjunctions and prepositions. FSCL features an LL2 grammar which is used to check coding sentences for syntactical correctness. The vocabulary for FSCL is defined by the user of the language. Each word belongs to one of six categories which have been specified based on the word classes used in the English language. The use of categories allows the combination of a flexible vocabulary with a generic grammar. Each coding sentence is linked to a segment of a multimedia behaviour recording. A coding sentence together with its multimedia segment is called a code instance.

The Flexible Structured Query Language, FSQL, is closely related to FSCL. FSQL supports the retrieval of code instances. The query facilities provided by FSQL are structured in several layers. The basic queries examine the contents of coding sentences by

```

query_matching_algorithm (semantic_tree_pattern, query_tree_pattern)
{
    continue = true
    match = false

    while (continue AND NOT match) do
    {
        find next position in semantic_tree_pattern which matches current position in query_tree_pattern
        do not pass any position in semantic_tree_pattern which has a higher level than current position
        in query_tree_pattern

        if (level match between semantic_tree_pattern and query_tree_pattern found) then
        {
            check for word match at both current positions in patterns

            if (word match found) then
            {
                if (end of query_tree_pattern reached) then
                    match = true
                else
                    advance position in query_tree_pattern
            }
        }
        else // level match not found
        {
            if (end of semantic_tree_pattern reached) then
                continue = false
            else
                backtrack in query_tree_pattern to closest position of same level as current position in
                semantic_tree_pattern
                keep position in sentence tree pattern
        }
    }

    if (match) then
        description sentence fulfils query
    else
        description sentence does not fulfil query
}

```

Figure 5.17: Query matching algorithm

taking advantage of the grammar structure built into these sentences. In the next layer basic queries can be combined with the Boolean operators OR, AND and NOT to Boolean queries. In the final layer FSQL makes use of the multimedia segments attached to the code instances. Queries can be formulated to examine time, position or sequence properties of these multimedia segments. These query options can be used to select code instances relating to behaviour recordings of different media types.

Chapter 6

Towards a Specification for a Multimedia Information System

In this chapter a concrete multimedia information system for the advanced support for the study of behaviour is introduced. The specification for this system, called **PAC**, concentrates on the main requirements of such a system, as identified in Chapters 4 and 5:

- Design of a *multimedia* system;
- Support for both *quantitative* and *qualitative* analysis techniques;
- Use of *flexible* and *structured* coding and query languages;
- Consideration of *domain context* and *knowledge building*.

Besides the requirements of such a system that arise from the application domain of the study of behaviour, the specification specifically considers the following aspects of good software design:

- Modular design;
- Use of a database management system;
- Human computer interaction considerations.

Additional to the considerations on software design the concept of ‘sets’ is introduced in this chapter. Sets are used to group analysis data which are stored in the database. The grouping of these data allows the specification which collection of data analysis steps or operations apply to. Throughout the previous chapters references to the concept of sets has been made to explain the relevance of the grouping of data for the behaviour analysis process. In this chapter, the use of sets, based on the modular structure of PAC, is explained in detail.

In NVivo, the new version of the software NUDIST, sets have been introduced to group data and to target operations to the members of a set (Richards, 1999; Fraser, 1999) in a similar way as described in this thesis.

The specification outlined in the first four sections of this chapter is described at a conceptual level. Section 6.5 provides implementation specific details for the system PAC.

6.1 Modular Design

Dividing an application into modules has the advantage of splitting a complex application into smaller, more manageable parts. The modular design aims at achieving high cohesion and low coupling (Pressman, 1997). The module structure of the system has specific importance in the interaction with the database tables (Section 6.2) and in the interface design (Section 6.3). In the following paragraphs the modules of the system PAC are introduced. As there is a close relationship between modules and sets, the brief descriptions given for the modules make reference to the sets associated with the modules. An explanation of the sets is provided in Section 6.4. The modules of PAC are divided into the three groups of *Preparation*, *Analysis* and *Conclusion*.

The modules grouped under the term *Preparation*:

StudyFiles

The task of this module is to locate the files containing the behaviour recordings (video, audio, text and image data) on the computer's hard disk and to identify them for the domain by giving them a name and description. In context of the application PAC the behaviour recordings are referred to as *study files*.

Vocabulary

In this module the vocabulary to be used for the coding is developed. The vocabulary consists of words used to describe the behavioural events or concepts.

Studies

The Studies module contains the definition of the study. The study is given a name and description. Memos concerning the whole study are written. The study files and words of the vocabulary to be used in this particular study are identified.

The modules grouped under the term *Analysis*:

Description

This module allows the connection of a coding sentence (formulated using the vocabulary of FSCL defined for use in the study) or an annotation sentence (free-form text) with an exact location in a study file. The resulting descriptions can be stored in different sets to allow for separate description by different analysts or for descriptions in different analysis phases. (This module was initially called Description/Coding/Annotation in the PAC implementation.)

QueryLanguage

In this module the query language FSQL is used to retrieve thematic groups of code instances. It can be specified which coding set or sets a query is applied to. The query results are coding sentences relating to a specific position in a specific study file. They are stored in retrieval sets.

TextSearch

In this module searches for text strings in annotations are performed to form thematic groups of annotations. Again, it can be defined which annotation set or sets a search is applied to. The text search results are annotations relating to a specific position in a specific study file. They are stored in retrieval sets.

Counts/Calculations/Graphs

Based on the results of querying and text search, frequencies or durations can be calculated. These calculations, or simple counts of the number of result instances, are presented in graphs or tables which are then interpreted. Examples for these types of interpretation are the comparison of values to follow a specific behaviour over a period of time or the checking of intra or inter observer reliabilities.

Viewing/Comparison/ImageAnalysis

In this module selected sequences in the behaviour recordings, based on the results of querying and text search, can be studied. These sequences can be viewed in slow or fast motion, or parallel to other sequences. Graphical annotation tools can be applied to analyse image content.

ConceptBuilding

The coding vocabulary contains words for concepts. During the course of the study, relationships between concepts develop and become apparent. In this module these relationships between concepts can be recorded and presented in graphical format. The specified relationships can be considered for further queries performed with the query language.

The module under the term *Conclusion*:

Conclusion

In this module the results of the analysis steps are combined. The conclusion statement is formulated. The statement is supported by data from the analysis steps. These can be graphs, video sequences, text segments, or concept relationships. All these elements are

combined to form a presentation of the study finding, the conclusion. The conclusion is stored and can be referred to at any later stage.

An overview over the modules, in form of a screenshot from the application PAC, is given in Figure 6.1. The modules are presented in the logical sequence according to the behaviour analysis process. All modules of the application can be used in any sequence and any number of times with the natural restriction of some inherent dependencies like the need to set up a study before it can be used.

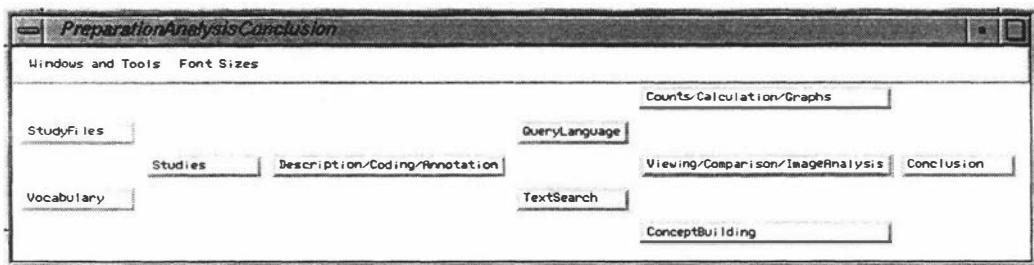


Figure 6.1: The modules of PAC

6.2 Interaction between Modules and Database Tables

All modules of the application access the database tables through a common mechanism, the *database interface handler*. The database interface handler provides procedures, accessible from all modules, which support the *select*, *insert*, *update* and *delete* operations on the database tables.

Each of the modules performs a range of tasks. To fulfil these tasks access to data stored in specific database tables is required. Each module has a number of database tables associated with it. This association is based on the type of access required. A table is associated to a module if this module requires write access to the table. Each table can only be associated with exactly one module, meaning that only this module can change the contents of the table. Each module can access any table in read-only mode. The association between database tables and modules for the application PAC is presented in Table 6.1. Appendix D contains detailed specifications for the PAC database tables in form of entity relationship diagrams.

Because the data of each table can be modified only in exactly one module it is possible to derive the data dependencies between the modules. Figure 6.2 shows how the tasks to be performed in a PAC module rely on data produced in other modules. The diagram shows that two of the PAC modules, the modules StudyFiles and Vocabulary, do not rely on data produced in any other modules. These two modules contain the tasks to be performed first

Modules	Associated Database Tables
StudyFiles	<i>study_files</i> : name and location of files containing behaviour recordings
Vocabulary	<i>words</i> : words of the vocabulary for FSCL and FSQL <i>structures, members</i> : hierarchy definitions for vocabulary <i>associates</i> : word link definitions for vocabulary
Studies	<i>studies</i> : definition of studies <i>memos</i> : memos for studies <i>study_study_files</i> : link of study files to studies <i>study_vocabulary</i> : link of vocabulary to studies
Description	<i>dca_sets</i> : definition of description sets <i>dca_sets_study_files</i> : link of study files to description sets <i>freeform_sentence</i> : annotations in description sets <i>structured_sentence, word_groups</i> : FSCL sentences in description sets
QueryLanguage	<i>ql_sets</i> : definition of query language sets <i>ql_sets_dca_sets</i> : link of description sets to query language sets <i>ql_set_items</i> : query results in query language sets
TextSearch	<i>ts_sets</i> : definition of text search sets <i>ts_sets_dca_sets</i> : link of description sets to text search sets <i>ts_set_items</i> : text search results in text search sets
Counts/ Calculation/ Graphs	<i>ccg_sets</i> : definition of counts, calculations and graphs sets including counts, calculations and graphs interpretation results <i>ccg_sets_ql_sets</i> : link of query language sets to counts, calculations and graphs sets <i>ccg_sets_ts_sets</i> : link of text search sets to counts, calculations and graphs sets
Viewing/ Comparison/ ImageAnalysis	<i>vci_sets</i> : definition of viewing, comparison and image analysis sets <i>vci_sets_ql_sets</i> : link of query language sets to viewing, comparison and image analysis sets <i>vci_sets_ts_sets</i> : link of text search sets to viewing, comparison and image analysis sets <i>vci_lists, vci_list_items</i> : viewing, comparison and image analysis interpretation results
ConceptBuilding	<i>cb_sets</i> : definition of concept building sets <i>cb_relationships</i> : concept relationships in concept building sets
Conclusion	<i>conclusion_sets</i> : definition of conclusion sets <i>conclusion_sets_ccg_sets</i> : link of counts, calculations and graphs sets to conclusion sets <i>conclusion_sets_vci_sets</i> : link of viewing, comparison and image analysis sets to conclusion sets <i>conclusion_sets_cb_sets</i> : link of concept building sets to conclusion sets <i>conclusion_set_segments</i> : conclusion data in conclusion sets
(Application wide)	<i>notes</i> : application wide notes <i>reference_files</i> : application wide reference files

Table 6.1: Association between PAC database tables and modules

in any study using PAC. The following modules, proceeding through the application from preparation to analysis and conclusion, all rely on data produced in earlier modules.

The information contained in Figure 6.2 is important for planning the interaction of the modules within the overall application. If the performance of a task in a module A is dependent on data produced in a module B, activities in module B will affect the outcomes of operations in module A. Because the data dependencies between the modules are known,

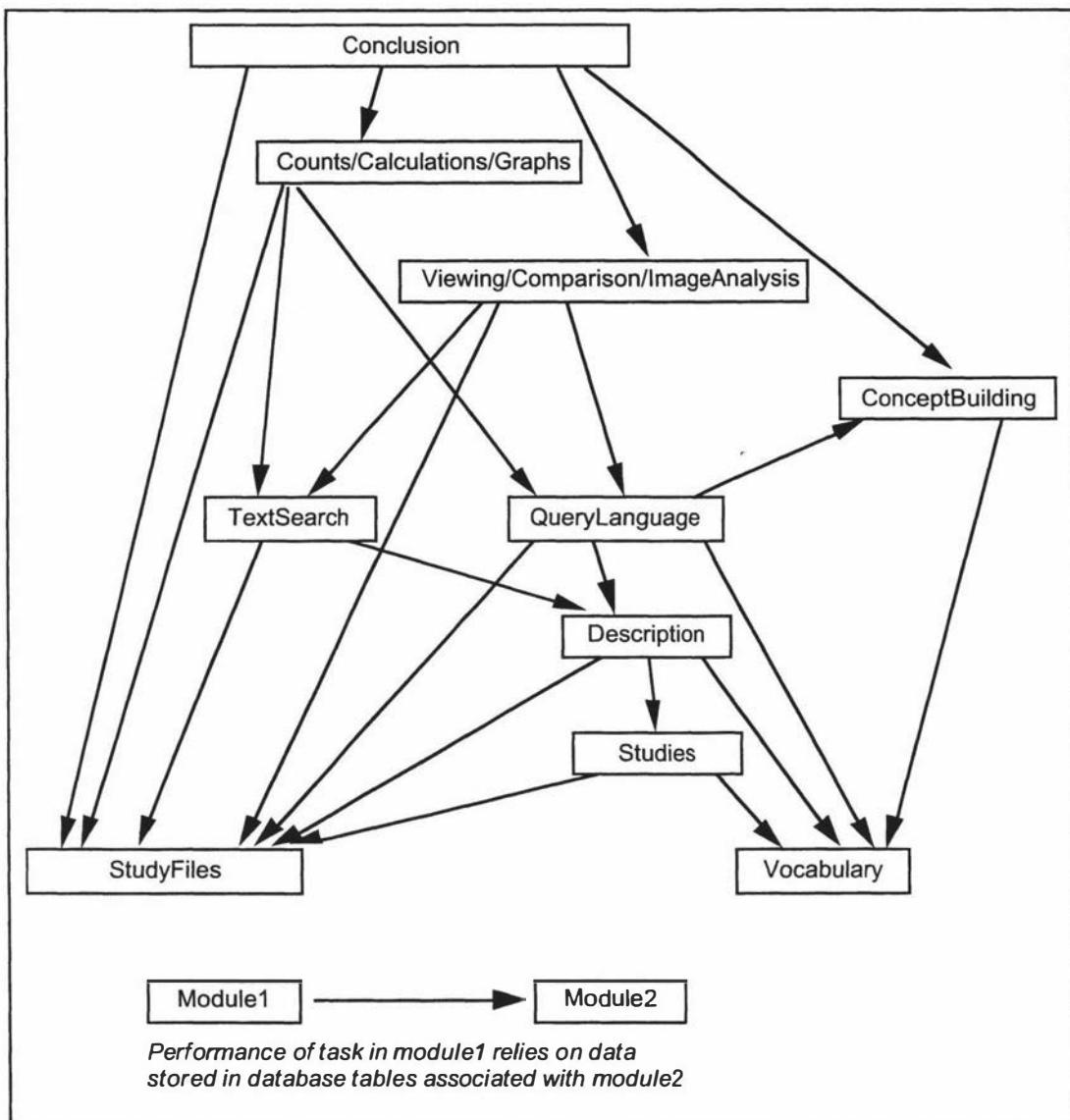


Figure 6.2: Execution dependencies between modules and database tables

it is possible to develop a strategy to prevent inconsistent use of data. This strategy can follow two possible directions:

- A module has to check for changes in relevant data administered by other modules.
- If a module modifies data it has to inform any other module depending on these data of these modifications.

Figure 6.3 shows another form of data dependency between the modules. In each module certain tasks are performed. These tasks produce results, stored in the database tables associated with the module. To look at these results, the database tables of the module

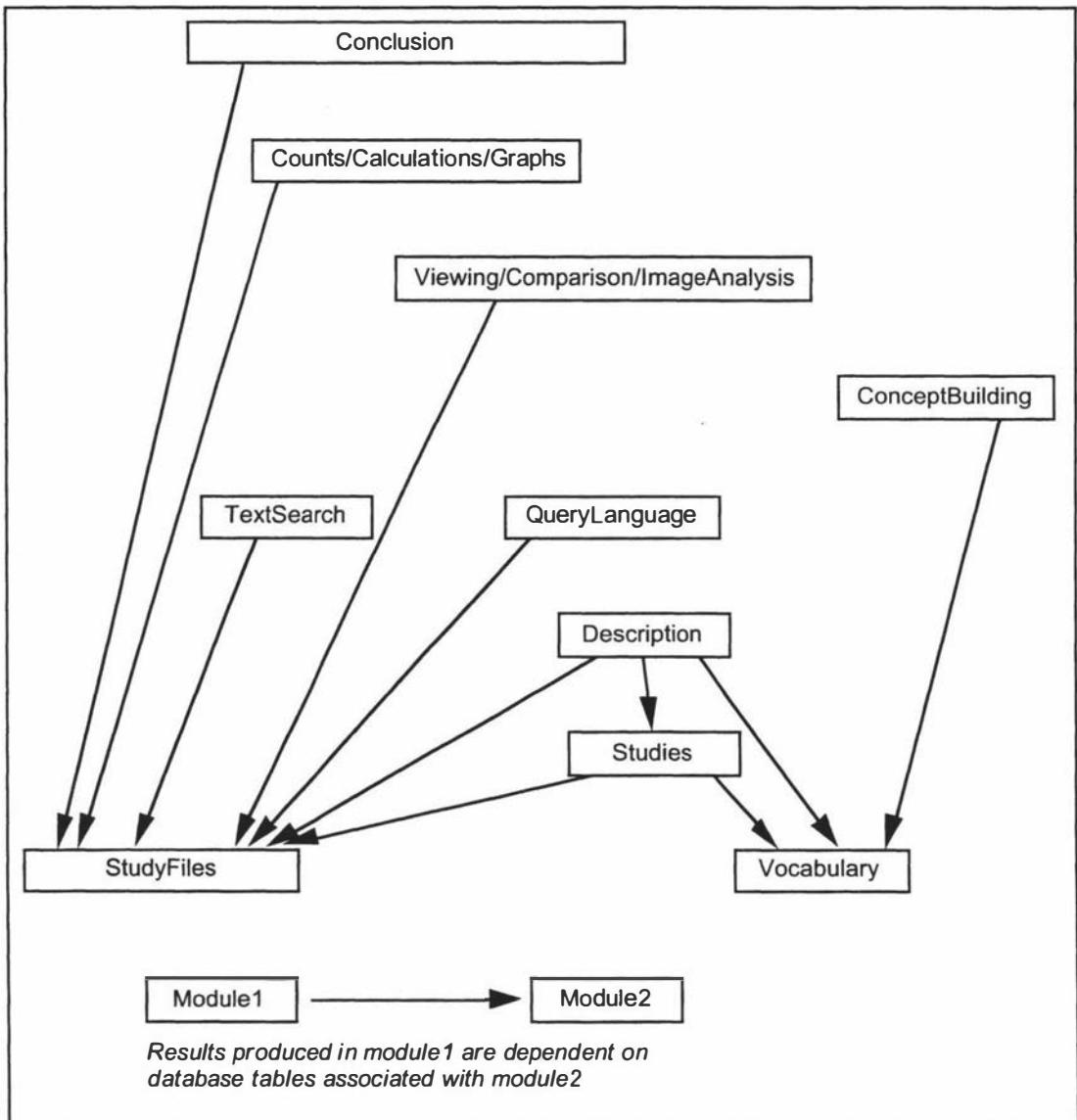


Figure 6.3: Result dependencies between modules and database tables

concerned and, in the case of all modules beside the StudyFiles and Vocabulary modules, database tables of additional modules have to be accessed.

A comparison of the two types of data dependencies, as indicated in Figures 6.2 and 6.3, shows, that fewer dependencies exist for the viewing of results than for the performance of tasks. The limited number of dependencies for the viewing of results in the QueryLanguage and Conclusion modules warrants special emphasis.

Accessing the results of queries performed in the QueryLanguage module is only dependent on the data produced in the StudyFile module for viewing the study file segments linked to

the query results. This specifically implies that the query results are independent from the code instances, produced in the Description module, and from the vocabulary, defined in the Vocabulary module (this is shown in Section 5.2.4). This independence means that existing query results can continue to be accessed and used in the interpretation steps even if changes are made to the vocabulary and the collection of code instances. The consequence of this independence is an increased flexibility in the way a study of behaviour can be performed.

As for the QueryLanguage module, the only data dependency for the Conclusion module in terms of accessing existing conclusions is to the StudyFile module. The significance of this lack of further dependencies can be expressed in two ways:

- As for the QueryLanguage module, this relative independence means that existing conclusions are not affected by changes in the interpretation modules.
- It would be possible to design a reduced version of a system which would only allow the browsing of existing conclusions without providing the facilities to perform analysis. Such a reduced system only needed to access the database tables associated with the Conclusion and StudyFile modules. These ideas are further extended in Chapter 8.

6.3 Interface Design using Module based Window Management

In most modules of PAC a variety of data has to be displayed. For the Description module, the study file showing the behaviour recording, the selected vocabulary, existing and newly created coding sentences and annotations, and the related study file segments have to be accessible. Yet the amount of data which can be displayed concurrently is restricted by the screen area available. A further requirement, specially in the area of qualitative analysis (Weitzman and Miles, 1995), is the need for flexibility in arranging interface elements and the choice of the user to display data important for the current analysis step. The difficulty in designing the user interface for the PAC application is multiplied by the fact that PAC consists of ten modules plus the PAC main application windows.

The literature shows different approaches for solving interface problems involving multiple windows by creating virtual work areas on application program level (Schroeter and Patrick, 1996) and on operating system level (Henderson and Card, 1986). In these approaches the screen space problem is solved by distributing the interface elements over several virtual work areas and providing the user with a mechanism to switch between these areas. The Henderson and Card approach is designed to support the interface requirements of separate, unrelated tasks and is therefore not suitable for an application like PAC where a strong relationship between the tasks performed in the different modules (like Vocabulary

and Description) exists. While this aspect is supported in the interface design by Schroeter and Patrick, the requirement for accessing interface elements from different modules (like the Description and Querying modules) at the same time is not facilitated.

To meet all the interface requirements of PAC concerning the display and management of interface elements, an interface design using *module based window management* has been developed. The data belonging to one module are spread over several windows of the following types:

- Each module has a main window which contains the tasks related to setting up sets required for the module task. The main windows of all modules always exist but can be hidden from the users view or can be iconised.
- Each module owns a collection of supplementary windows in which the tasks of the module are performed or supported. These windows can be closed, hidden or iconised.

The PAC main application window allows access to the main windows of all modules and manages two application wide windows for general notes and PAC status messages. To facilitate the performing of tasks of closely related modules the windows of several modules can be displayed on the screen at the same time. The user controls which individual windows are displayed in which size and position on the screen. A schematic overview over the window structure on a module basis is given in Figure 6.4.

To assist the user in managing the different windows belonging to the different modules, the windows are coordinated on a module basis. Each main module window provides menu options for the following management tasks:

- Close, hide or iconify all supplementary windows belonging to the module with one command. This allows the user to tidy-up the screen with one command on a module basis. When the supplementary windows are displayed again, after having been hidden or iconised, they appear with the same content and in the same position or size they had been displayed before. Opening a supplementary window after it had been closed opens the window in a default format.
- Hide all windows belonging to a module. With this command the user can hide all windows belonging to the module, including the main module window, with one command, which is useful for swapping between different modules. If the module is accessed again via the PAC main application window all module windows are restored to the status they had been in before hiding.
- Reset a module. Using this command all supplementary windows belonging to the module are closed. All resources occupied by the module, like memory intensive video players, are released. The data in the main window of the module are reset and the main window is hidden from the users view.

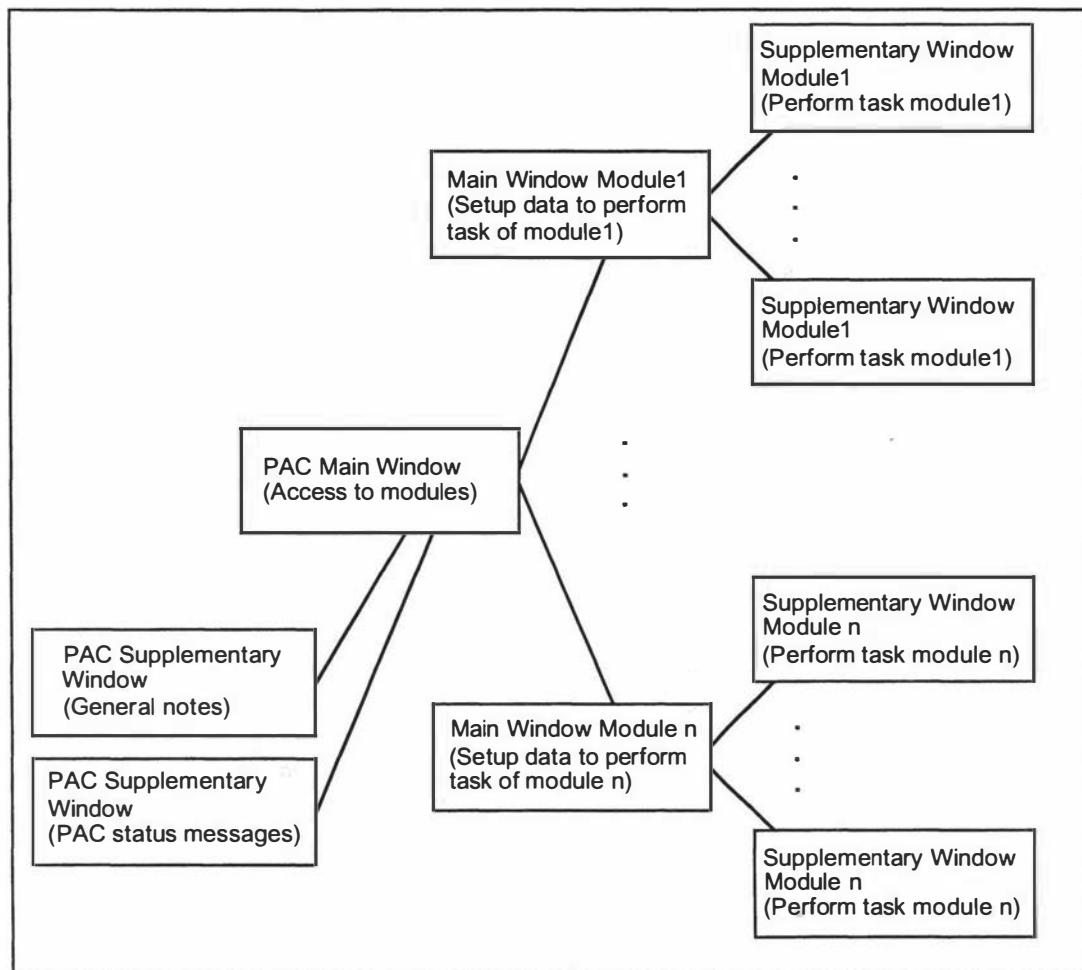


Figure 6.4: Module based interface elements in PAC

In summary, the approach introduced groups the interface elements on module basis, gives the user full control over which interface elements are to be presented and supports the user in managing the interface elements on a module basis. Appendix C lists in detail the windows belonging to each module.

6.4 Use of Sets

Sets are used to group data in the analysis steps of a behavioural study. The sets define which data are available for a behavioural analysis step and what operations are possible in consecutive analysis steps. Sets are defined for the following modules of the application PAC:

- *Module Description*: Description Sets (called DescriptionCodingAnnotation Sets in the PAC implementation);
- *Module QueryLanguage*: QueryLanguage Sets;
- *Module TextSearch*: TextSearch Sets;

- *Module Viewing/Comparison/ImageAnalysis:* VCI Sets;
- *Module Counts/Calculation/Graphs:* CCG Sets;
- *Module ConceptBuilding:* CB Sets;
- *Module Conclusion:* Conclusion Sets.

A set definition has two parts:

- Descriptive information like name and textual description of the set which is common to all different types of sets;
- Information which is specific for each set type and defines the range of data the set operations apply to:
 - Description Sets: Study files the descriptions can be applied to;
 - QueryLanguage Sets: Description Sets the queries apply to;
 - TextSearch Sets: Description Sets the text search applies to;
 - VCI Sets: QueryLanguage and/or TextSearch Sets the VCI operations apply to;
 - CCG Sets: QueryLanguage and/or TextSearch Sets the CCG operations apply to;
 - CB Sets: The words of the Concept category of the vocabulary;
 - Conclusion Sets: VCI, CCG, CB Sets which contribute to the conclusion.

Figure 6.5 shows how the different sets are connected by providing the input data for modules following in the analysis process.

The concept of sets has been introduced to allow the user to group description and interpretation data according to purpose. With data stored in different sets it is possible to access the data separately if necessary. With the possibility of combining several sets most operations can still be applied to a range of data beyond set boundaries. There are a number of considerations which influence the distribution of data across sets in the different analysis steps:

- Some quantitative calculations like duration or frequency can only be sensibly performed if they refer to only one study file (see Section 4.5). The sequence of description, retrieval and interpretation sets to perform such calculations has therefore to be constructed in a way relating only to a single study file.
- Most quantitative measures are only meaningful if they are based on strictly counting each behavioural event only once. The sequence of sets used in producing the measures has to ensure that observations by different observers, descriptions of different level of detail, or overlapping descriptions for single behavioural events are separated into different sets.
- Descriptions by different analysts or observers can be separated into different sets to facilitate inter-observer reliability checking in a quantitative sense or the comparison of descriptions of different analysts in a qualitative sense.

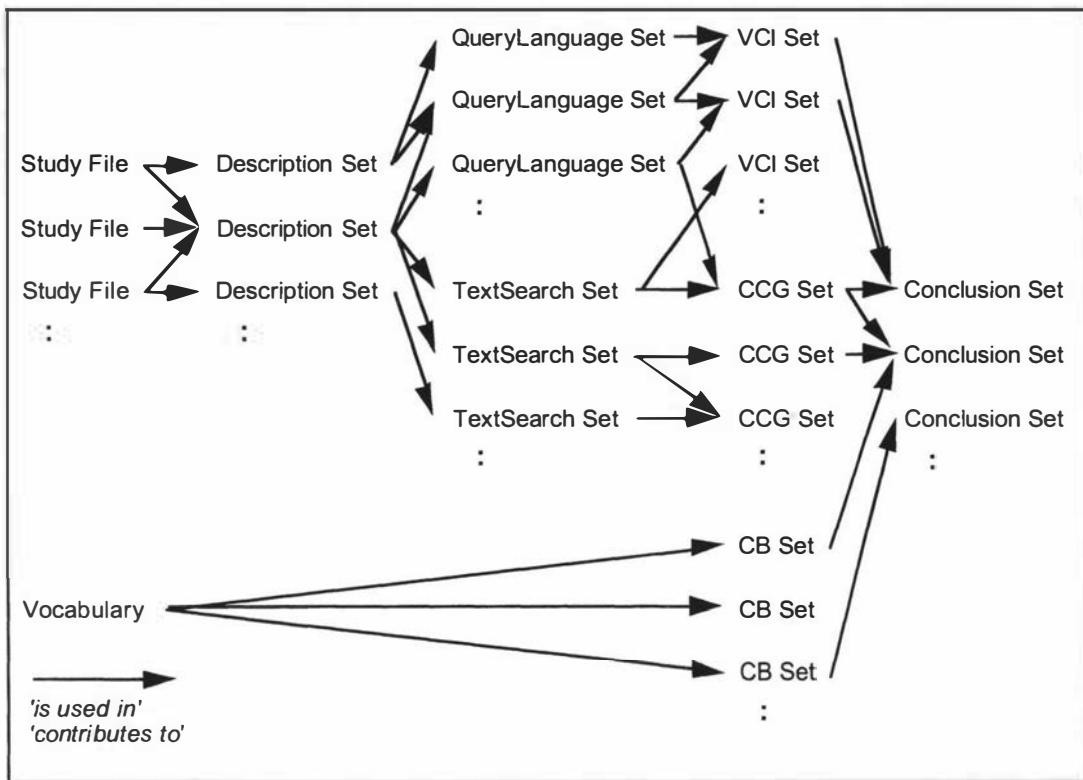


Figure 6.5: Role of sets in the analysis process

- Sets can be used to separate data of different phases of a study. This can refer to different observation or treatment phases in quantitative analysis or to a modified use of a coding scheme with growing understanding of the issues involved in qualitative analysis.
- The sets have to be defined to give access to the range of data required for an operation. If the behavioural descriptions applied across several study files are to be combined in the retrieval phase then the retrieval set has to provide access to all these descriptions. If a quantitative interpretation requires the comparison of several measurements the CCG set has to provide access to the relevant measures.
- If a study emphasises the sequence of behavioural events the storage of descriptions belonging to different ‘threads’ in separate sets facilitates the uninterrupted presentation of descriptions in temporal ordering.
- Within one study analysis might be performed following two different analysis directions. Data belonging to the different directions can be stored in different sets and possibly compared in the conclusion of the study.
- Sets can be used to facilitate a convenient management of the analysis data. The amount of data stored within each set has to be balanced to the number of sets required.

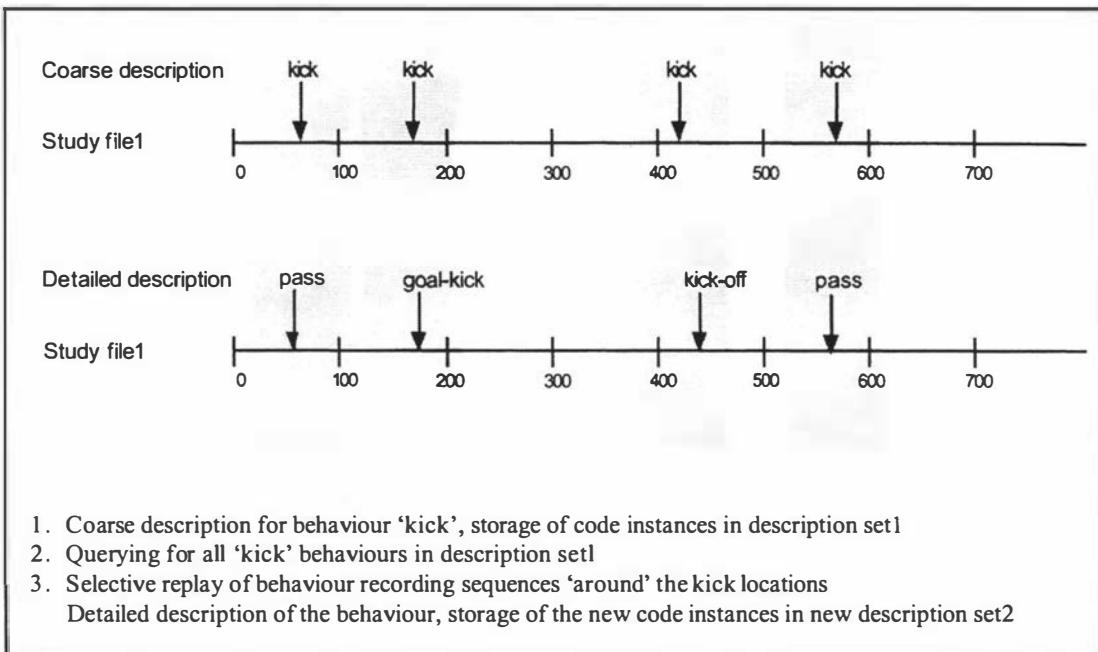


Figure 6.6: Potential coding - querying - coding sequence

Some of the above considerations are illustrated in the following figures. Figure 6.6 shows a potential coding - querying - coding sequence where sets are used to separate an initial coarse description from a subsequent selective detailed description. The use of different sets in this example helps to prevent counting the same behavioural event twice. In Figure 6.7 a simple analysis sequence aimed at quantitative analysis is presented. Two description sets are used to capture the same type of behaviour in different study files. The behaviour descriptions are retrieved and stored separately in two query language sets. The data in the two query language sets are used to compare quantitative measures relating to the two separate sets of data. Figure 6.8 indicates the combination of description data across several study files in one query language set. The behaviours in three different study files are described separately in different description sets. One query language set is defined to access data of all three description sets. The results stored in the query language sets are to be used to study the related behaviour recording segments in a qualitative style interpretation. All three example scenarios presented in this section relate to the study of a team sport like soccer and have been presented in a very simplified form to highlight specific aspects.

The concept of sets facilitates the combination of quantitative and qualitative analysis methods within one study:

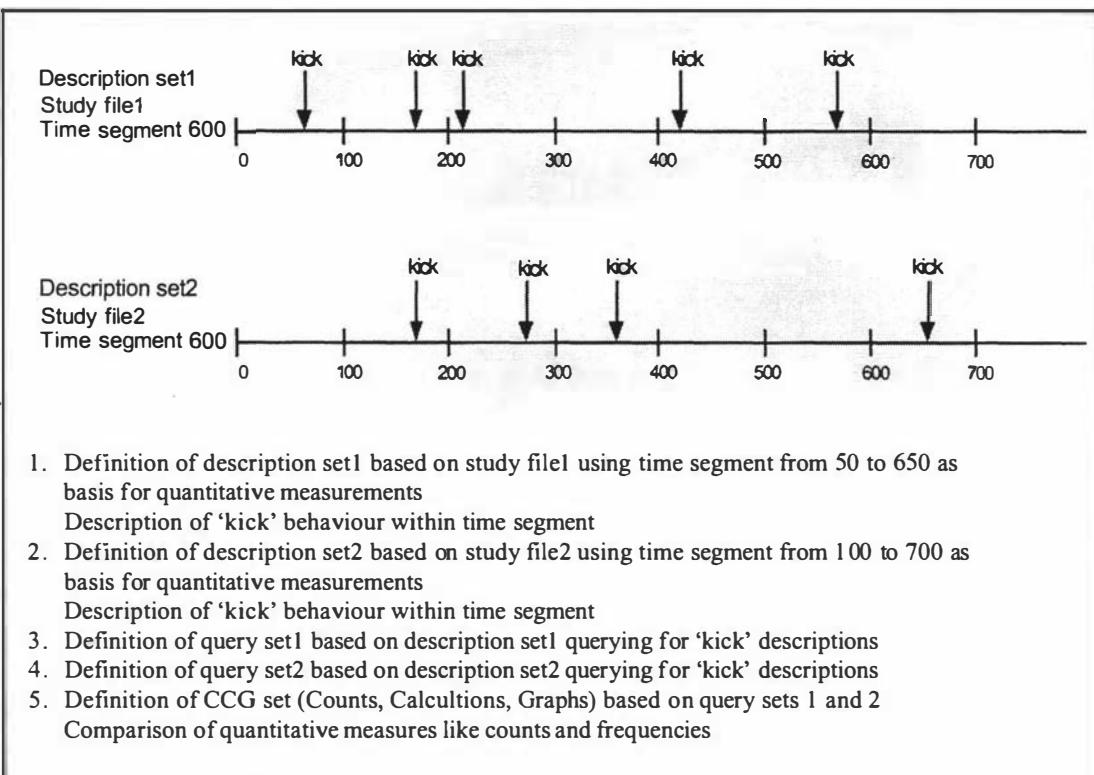


Figure 6.7: Potential analysis sequence aimed at quantitative analysis

- The separation of data into different sets supports the calculation of precise quantitative measurements.
- The application of analysis steps across data of several sets supports the combination of data relating to different study files, observations, descriptions, or retrievals performed by different analysts.

Figure 6.9 shows an example of using sets to support quantitative and qualitative analysis steps within one study.

6.5 The Implementation of PAC

Emphasis of the Implementation

In conjunction with the arguments for implementing a prototype the following emphasis for the development of the application PAC was set:

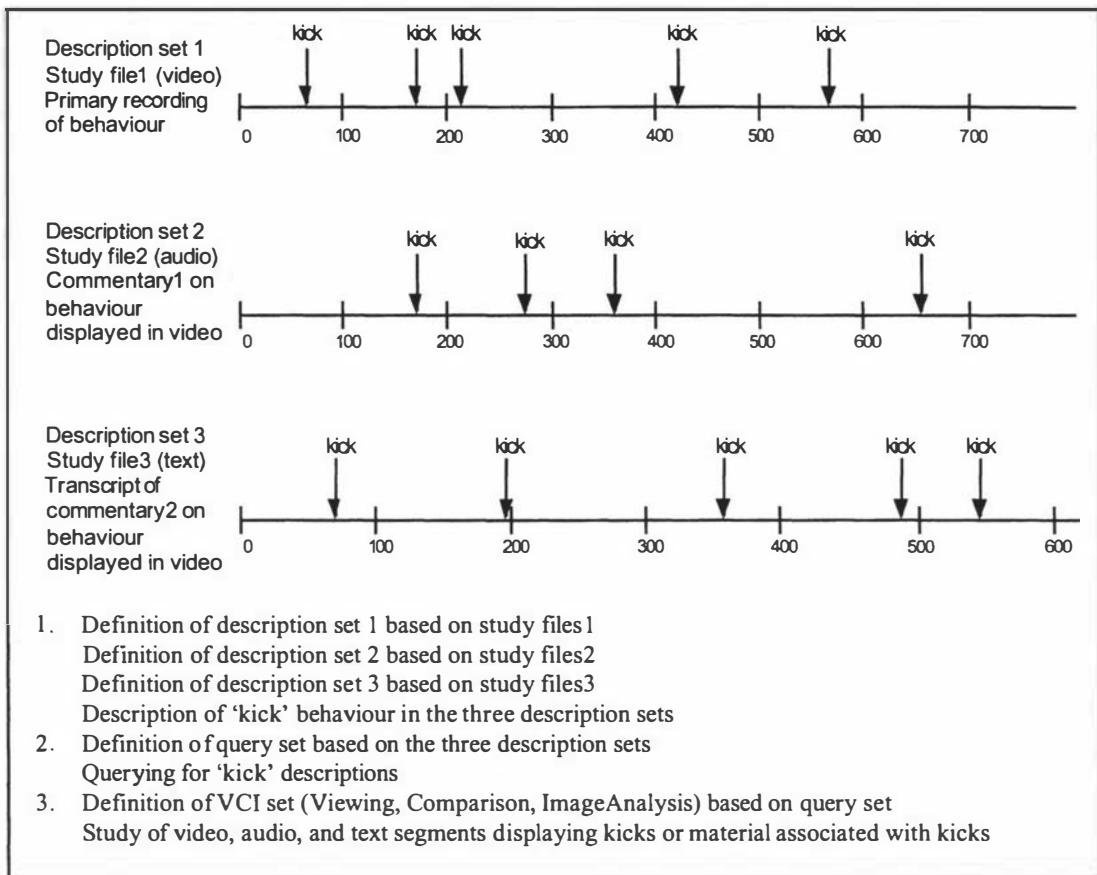


Figure 6.8: Potential analysis sequence aimed at qualitative analysis

- The implementation had to support all main components of the conceptualisation for an advanced multimedia information system to support the study of behaviour. These components were the support of digital behaviour recordings of multiple media types, the combination of both quantitative and qualitative analysis methods, the new coding and query languages, the support of a behavioural study from preparation to analysis and conclusion, and the placing of a behavioural study in a domain context.
- The implementation had to be advanced enough to make it possible to conduct case studies in conjunction with domain experts.
- Besides supporting the major components of the concept, the implementation was to at least indicate minor components of the concept to highlight further possibilities and to serve as a starting point for discussions with the domain experts.

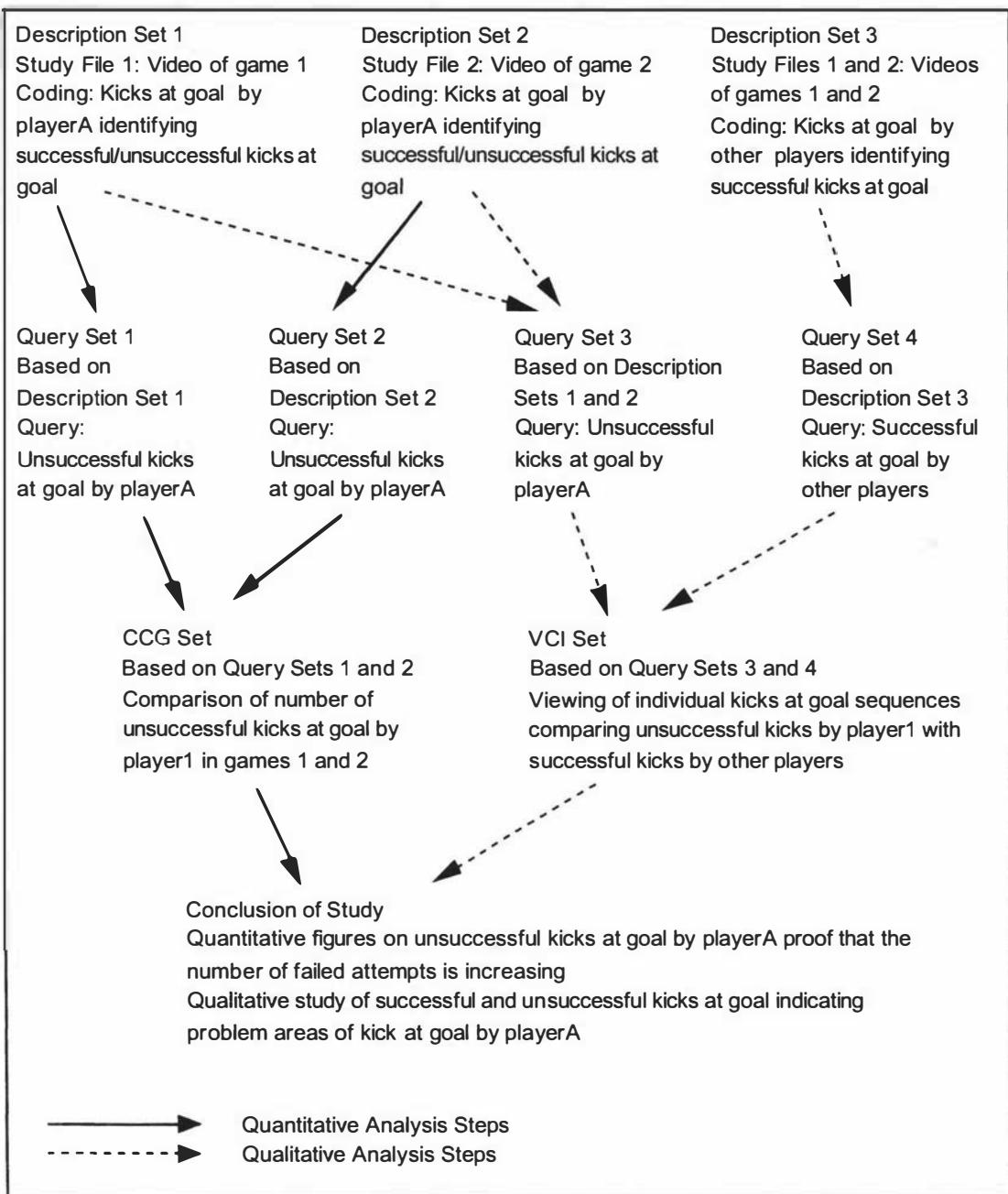


Figure 6.9: Use of sets to combine quantitative and qualitative analysis

What was Left out of the Implementation

Due to limited time resources some elements necessary for a full implementation were neglected. These elements were:

- The only media types currently supported are video (consisting of picture and sound information) and text. Audio and image data have not been included.
- Behaviour recordings can refer to a common timeline (see Section 4.1). This is not considered in this implementation.
- Image analysis via graphical annotation has not been implemented sufficiently.
- The application does not warn the user of a potential loss of data by resetting or quitting modules.
- Only a limited number of error messages are displayed.
- There is no help system or reference manual.
- The correct sequence of deleting data from the database for dependent data is the responsibility of the user.
- There is no user identification for access to the analysis data or behaviour recordings.

Implementation Platform

The application was implemented using the Java programming language, the Java Media Framework (JMF), and the relational database Msql. It was developed on a Silicon Graphics Indy unix workstation.

The programming language Java was chosen because of the potential benefits of its platform independence. The prototype could be developed, tested and initially used on the fast Unix platform and, at a later point-of-time, be easily ported to high-end multimedia PCs. JMF allowed an easy integration of multimedia data access into a Java program. As an object-oriented programming language Java was well suited to create an interactive and modular application. A final reason for selecting Java was the free availability of Java via the internet, including the JMF and a basic development environment consisting of compiler, interpreter and debugger.

A relational database was required to store the analysis data. As it was not necessary for the application to store the behaviour recordings in the database, a relative simple relational database was sufficient. The Msql database was selected because it was freely available via the internet to research organisations, because it could be easily accessed from the Java program via the Java Msql classes, and because a copy of the database server could be installed on the development system, thus creating independence from departmentally administrated database servers.

In detail, the following hardware and software versions were used:

- Silicon Graphics Indy workstation, operating system Irix 6.2;
- Silicon Graphics Java 2.1, based on Sun Microsystems Java Development Kit 1.0.2;

- Silicon Graphics version of the Java Media Framework 0.95;
- Hughes Technologies Msql 2.0.3;
- Msql Java classes 1.2.8 by Darryl Collins.

To support the preparation of material for the case studies additionally an analog video recorder and a Cosmo compression card were available.

The use of this very early version of JMF brought with it some restrictions in functionality for the application. The recording of audio, as suggested for the conclusion recording, was not possible. Frame level access to video recordings was not available and had to be replaced by time based access. The correct video image was not displayed after repositioning the video without playing it. The replay of video occurred relatively slowly because no access to the hardware compression card was possible via a JMF video player. While the pure software replay was welcomed because of the ability to replay two or more videos in parallel, the faster performance of a hardware assisted replay would have been of advantage for some aspects of the case studies. The video replay was, due to performance restrictions, limited to medium quality half frame size videos.

The above mentioned limitations of JMF were discussed in several email contributions to the jmf-interest@SUN.com mailing list in the years of 1997 and 1998. A number of the shortcomings of JMF were to be addressed in the JMF 2.0 release of Sun Microsystems, announced for mid 1999.

Development Process

After the conceptualisation of a new multimedia information system supporting the study of behaviour had been finalised and discussed with experts from different behaviour analysis domains, a specification for a prototype implementation was developed. Separate tasks within the conceptualisation were identified and the whole application divided into modules according to the tasks. The interaction between the modules and the internal structure within the modules were specified. The database tables were designed. The access and modification rights of modules for groups of database tables were identified. The formal specification for the coding and query languages were formulated. The coding and query processes were defined. General guidelines for the user interface of the application were produced which comprised the styles of individual windows and the combination of windows within modules. A prototype was developed according to the specifications over a period of about eight month.

This prototype was then evaluated for criteria like accessibility of multimedia data, database access, interaction of modules, and usability. The functionality of the prototype in

supporting the tasks identified according to the concept design was tested. With the information gained from evaluating the initial prototype and in the light of some further developments to the overall concept it was decided to substantially redevelop the prototype. This was done over a time frame of about four month. From there on only minor extensions and corrections, mostly prompted by conducting the case studies, were performed.

Chapter 7

Case Studies

In Chapters 4 and 5 of this thesis a new framework to support the study of behaviour is proposed. In Chapter 6 the information system PAC is introduced which provides a prototype implementation for this framework. The case studies described here were performed to investigate the claim that the ideas introduced in the framework provide advanced support for the study of behaviour. The implementation PAC was used to conduct the case studies.

Employing the research method of case studies was seen as most appropriate because it allowed a number of behavioural domain experts to gain intimate knowledge and understanding of the concepts suggested for a new multimedia system to support the study of behaviour. The domain experts would use PAC to analyse their behavioural data. Based on the analysis results achieved, the feedback given by the domain experts, and the observations made by the author during the duration of the case studies, the usefulness of the framework suggested could be evaluated. It was further assumed that the domain experts, through the intensive work with PAC, would achieve enough familiarity with the concepts of PAC to be able to comment of the usefulness of PAC beyond the limitations of both the specific studies and the prototype implementation.

The emphasis throughout the case studies was on evaluating the concepts behind the implementation of PAC. While the implementation made it possible to conduct the case studies its main purpose was to facilitate the application and analysis of the concepts. To enhance its readability this case study report mostly refers to 'PAC' instead to of using a construct like 'the concepts behind PAC'.

7.1 Case Study Design

The next sections describe the research design for the case studies following the recommendations in Yin (1993). First the case study question and the more detailed propositions are formulated. This is followed by a discussion of the selection of cases for the multiple case design. The data collection and interpretation steps for the individual case studies are outlined and followed by a section on cross-case evaluation.

7.1.1 Case Study Question

In this research a framework for a multimedia information system to support the study of behaviour is introduced. This framework contains features like the structured yet flexible coding and query languages (FSCL and FSQ), the use of digital media, and the support of quantitative and qualitative analysis techniques. The claim is made that implementing a computer application according to this framework leads to **advanced** support for the study of behaviour. Based on this claim the two main research questions for the case studies were formulated:

‘Can PAC be used to support the study of behaviour?’

‘Does PAC provide advanced support for the study of behaviour?’

The main research questions were divided into a number of propositions to be evaluated:

- Usefulness of the flexible structured coding and query languages, FSCL and FSQ;
- The advantages of digital as compared to analog video;
- The support for combining quantitative and qualitative research methods;
- The advantages of the combined analysis of behaviour recordings of multiple media formats;
- The benefits of a multimedia conclusion construction;
- The possibility of building up a domain knowledge base.

Some of the ideas introduced with PAC could not be fully tested within the scope of these case studies. For example, building up domain knowledge within a behavioural domain can only happen over time by performing a series of studies. This was not possible within the scope of this research and therefore the domain experts could only be asked to comment on the usefulness of this idea based on their expertise.

Further restrictions arose from the fact that PAC is a prototype implementation. As such PAC has some limitations regarding the number of features available to the user. The quantitative interpretation tool, for example, offers only basic descriptive statistical functions and not the range of functions a user would expect from a full implementation. The tool for the conclusion construction is not implemented to the extent suggested in the theoretical description of the concepts which was due to the technical limitations posed by the development environment (see Section 6.5).

7.1.2 Multiple Case Design and Study Selection

A multiple case design was chosen to produce more compelling, richer evidence compared to performing only a single case study (Yin, 1994). The design logic aimed in the first instance at theoretical replication (Yin, 1994; Rose, 1991) and in the second instance at

diversity (Rose, 1991). The research question posed for each single case study was to find evidence supporting or contradicting the claim that the proposed framework would provide advanced support for the study of behaviour. The combination of multiple case studies was to provide replication of the findings. PAC has been designed to support the study of behaviour in a wide range of behavioural domains utilising both quantitative and qualitative analysis methods. Selecting case studies representing a range of behavioural domains and research methods was to show the suitability of the framework for a diversity of studies.

All three case studies selected were built around existing studies of behaviour. In one study PAC was used to analyse the behavioural data according to the analysis plan already defined for the study. In the two other cases PAC was used to add an exploratory line of analysis to the existing studies.

There were a number of advantages of building on existing studies. Using existing studies, which had been designed independently from PAC, would help to answer the first research question, showing that PAC was a suitable tool to support the study of behaviour. Applying PAC to their existing studies made it easier to motivate the domain experts to invest a considerable amount of time into the understanding of PAC. The domain experts were interested in advancing their studies and recognised PAC as a tool which could help to further their goals. Another advantage of using existing studies was a considerable time saving compared to setting up new studies. In all three studies selected the behavioural data had already been collected and the data analysis could start.

For two reasons the three case studies were performed in parallel. Firstly, the total duration of each study was about one year and it was therefore, in the given timeframe for the whole research, necessary to perform the studies in parallel. Secondly, with the aim of replication, the case studies were supposed to evaluate the same theoretical framework and therefore could easily be conducted in parallel.

7.1.3 Case Study Evidence

Burns (1997) lists four sources of case study evidence, namely documents, interviews, participant and non-participant observation, and artefacts, which all have been used in the case studies described.

The case studies were performed in close interaction between the domain experts and the author. Numerous meetings were held to prepare the case studies and discuss analysis steps. The author produced summaries of these meetings and distributed them to the domain experts. These summaries captured the ideas discussed in the meetings and outlined the developing analysis steps. In one case study all preliminary discussions were conducted by

email. In this case complete documentation was available by nature of the communication mechanism.

The application program PAC was initially operated by the author, demonstrating its use to the domain experts. Step by step the domain experts learned to use the program themselves until they became relatively independent users of the main functions of PAC. During the use of the program PAC to perform the analysis of the behavioural data, the role of the author changed from participating observer to non-participating observer. In the initial phases of working with PAC the domain experts needed a relatively high amount of guidance. The author not only operated PAC but also suggested which features of PAC could be applied to a specific analysis task. This occurred mainly in the phase of the case studies where the domain experts were still finalising their analysis steps. By the time these steps had been finalised, the domain experts had become familiar enough with PAC to operate the program largely by themselves. All analysis of the behavioural data was performed solely by the domain experts. The author remained present while the domain experts were operating PAC, but was mainly there to give technical advice in using PAC and to correct any problem which might occur considering the prototype status of PAC.

The analysis with PAC produced a number of artefacts. These were listings of the vocabulary, the description sentences, the queries and query results, information about the set structure chosen for the studies, and graphical presentations derived from query results. All these artefacts documented the way PAC was used to support the studies of behaviour.

After the domain experts had finished their studies, semi-structured interviews were conducted. The domain experts were asked to both comment on their experiences of using PAC in the concluded studies and in relation to potential future studies in their areas of expertise. Figure 7.1 shows the questions the interviews were based on.

7.1.4 Analysis of Evidence

The analysis of the case study evidence was aimed at providing answers to the case study propositions and, based on these, the main case study questions. To prepare the analysis, the audio recordings of the interviews were transcribed. The documents, artefacts and notes taken during the observations were sorted and reviewed. Table 7.1 shows which main sources of evidence were used to answer the propositions.

The evidence not used directly to evaluate the propositions contributed, together with the findings for the propositions, to answering the main case study questions. This was evidence contained in the documents, artefacts, and observation notes as well as discussions following some of the interview questions. Questions 4 and 8 aimed at a wider context beyond the limitations of the current case studies, whereas questions 9 and 10

Questions to the Domain Experts

These questions are asked in the context that you have conducted a study of behaviour using PAC. Please answer the questions based not only on your experiences of using PAC for your current study but in context of the range of studies you have performed or plan to perform.

1. PAC supports the combined analysis of behaviour recordings of several media types (video, text). How can you make use of this feature in your range of studies?
2. What impact do the concepts of the PAC coding language and query language have on your studies of behaviour?
3. Did the development of the vocabulary help you to gain a better understanding of your data or planned analysis steps?
4. Does using PAC influence your method of analysis? If yes, how?
5. Two of the concepts PAC uses to support both quantitative and qualitative analysis methods are:
 - the flexible and structured coding language/query language to allow both rich and precise descriptions of behaviour
 - the provision of interpretation tools to view behaviour recording sequences, to perform basic calculations/comparisons, to study the extracted behaviour descriptionsDo you agree that these concepts facilitate the use of both quantitative and qualitative analysis methods?
6. The PAC implementation indicates how the presentation of the conclusions of a study can be supported by analysis data (graphs, behaviour recordings, ..). Can you see how you could use this feature?
7. The concept behind PAC suggests that domain experts could over time, by conducting several studies using PAC, build up a collection of 'domain knowledge' consisting of behaviour recordings, vocabulary, analysis data and conclusions. Do you think that this would be useful?
8. How would you use PAC for further behaviour analysis studies?
9. Would you like to make further comments on any aspect of the concepts of PAC?
10. Would you like to make any comments on the strength or the weaknesses of the implementation of PAC?

Figure 7.1: Questions to the domain experts

prompted the domain experts for general comments on any of the aspects of concept and implementation of PAC. The case study reports for the individual studies were presented to the respective domain experts for comment and verification. The feedback from the domain experts was integrated into the reports.

7.1.5 Cross Case Evaluation

The cross case evaluation followed two directions in accordance with the multiple case design logic looking for replication and diversity. To check for replication the results of the

Proposition	Main Source of Evidence
Usefulness of the flexible structured coding and query languages, FSCL and FSQL	Questions 2 and 3
The advantages of digital as compared to analog video	Observations and discussions during case studies
The support for combining quantitative and qualitative research methods	Question 5
The advantages of the combined analysis of behaviour recordings of multiple media formats	Question 1
The benefits of a multimedia conclusion construction	Question 6
The possibility to build up a domain knowledge base	Question 7

Table 7.1: Propositions and main source of evidence

individual studies were compared on a proposition and main case study question level. Additionally the single cases were investigated in regard to diversity. Characteristics of the individual studies were identified with the aim to highlight different needs for support.

7.2 The Individual Case Studies

All three case studies were largely performed according to a common format. The first phase was the familiarisation phase which was a two-sided process. The behavioural domain experts had to become acquainted with the concepts and implementation of PAC. The author had to understand the behavioural studies the domain experts were working on to be able to suggest possible uses of PAC. Following on from the first familiarisation the scope of the case studies was defined. The needs of the domain experts to gain some benefit for their own studies and for the author to test the propositions had to be balanced. Time and resource restrictions had to be considered. The next phase consisted of technical preparations for the studies like setting up necessary structures in PAC and digitising the analog video recordings. From there the development of the PAC vocabulary followed. As will be seen in the individual case study reports this step required considerable time because the vocabulary is closely linked to the concepts of the coding and query languages and therefore central to the main concepts of PAC. In the next steps, the behaviour recordings were coded and the resulting descriptions queried.

Following the exploratory nature of the case studies, all the phases described so far did not occur in a strict linear sequence. A number of the issues were addressed repeatedly. The growing understanding of the domain experts for PAC raised new questions and produced new ideas of how to use PAC to advance the studies of behaviour.

As the practical work with PAC had come to some form of conclusion and the domain experts had developed a good understanding of PAC, the semi-structured interviews were

conducted. The domain experts were asked to report on their experiences of using PAC in their current studies and on the prospects of using PAC for other studies in their fields of expertise.

7.2.1 The Reading Groups Case Study

The Reading Groups case study was based on a study titled 'Contexts of Support: Learning to Read in Small Groups in New Zealand' which builds on work by Wilkinson and Townsend (1999). The purpose of this study was to analyse the parameters which contribute to learning to read in New Zealand primary schools. The context of the study was strong criticism of teaching reading in small groups that have been formed based on the ability of the students. Overseas research suggests that this kind of grouping disadvantages weaker readers. This study attempted to show that group tuition is successful in the New Zealand context, where the group membership of students is very flexible and the instructional practices of teachers may overcome the potential disadvantages for the weaker readers.

The study entailed a detailed analysis of the instructional procedures of selected "best practice" teachers and the reading abilities of a number of selected students over a year. Three different types of data were collected: video recordings of reading lessons, audio recordings of post-lesson stimulated recall interviews with the teachers, and results of word knowledge tests given to the students before and after the reading lessons. The study design included three different forms of coding. The video recordings were to be coded for 'word learning opportunities', describing every instance where one of the target words (as defined by experienced teachers who were not involved in the data collection) featured in the lesson in any form. In addition, the video recordings were to be coded for 'procedural engagement', capturing the attention status of each student. The third form of coding was to be based on transcripts of the interviews with the teachers. These interviews were to be coded for 'substantive engagement' of the students in the lessons according to the statements made by the teachers as they watched the videotapes of the lessons. The descriptions resulting from the three different forms of coding were to be combined and interpreted in the context of results from the word knowledge tests.

7.2.1.1 Familiarisation with PAC and Scoping of Case Study

The familiarisation of the domain expert with PAC occurred over a period of nearly half a year in the form of email messages and electronic exchange of documents. The domain expert was located about 600km from the site where PAC was implemented. The prototype status of the PAC implementation and its hardware requirements made it impossible to transport PAC to the user.

Initially descriptions of the behavioural study and general descriptions of PAC were exchanged. The behavioural study was well defined and a number of parameters for the analysis had been specified. Yet for one central part of the analysis, the analysis of the word learning opportunities, neither theory nor analysis steps had been finalised. These were to evolve in the course of the study.

Utilising the initial project description, the domain expert was presented with suggestions of how PAC could be used to facilitate his analysis. These suggestions started the exchange of a series of documents with the aim of developing a better understanding of the behavioural study on the one hand and of PAC on the other.

Once the conceptual issues had been clarified and the domain expert could see the value of using PAC for his study, he agreed to visit the PAC site. Some behaviour recordings were presented and initial vocabulary defined. The domain expert was given a demonstration of the implementation of PAC. The vocabulary was further developed and the first coding was done.

At the end of the visit, the domain expert decided that the value of using PAC would compensate for the difficulties and costs associated with travelling to the site. He decided to use PAC to perform the analysis of a subset of his behaviour recordings and to concentrate on the coding of word learning opportunities, the area where PAC could provide his study with the biggest advantage.

The extensive exchange of written information, as compared to face-to-face meetings, proved beneficial in helping the domain expert and the author to think through the issues involved. Similarly, first addressing the conceptual aspects of PAC, then the implementation issues, proved beneficial in helping the domain expert to understand the capabilities of the system.

7.2.1.2 Process of Conducting the Case Study

The domain expert selected a subset of eight reading lessons for analysis with PAC. The video recordings of the lessons were digitised and the transcripts of the interviews imported as text documents. Then PAC study files were defined. Next, the vocabulary was discussed (as far as this was possible at the beginning of the study) and entered into PAC. Following from that description sets were defined.

The domain expert had decided to concentrate on the coding of word learning opportunities during his visits to the PAC site. The coding for procedural and substantive engagement was to be done by the domain expert without using PAC. The main reason for that was the need to travel to be able to use PAC and, related to that, the limited amount of time available

for using PAC. A second reason for performing the coding of the procedural engagement outside of PAC lay in the limited quality of the digital video replay and, at that point of time, in some shortcomings in PAC's ability to support coding of state changes in behaviour.

At the end of each visit to the PAC site, the domain expert was given files containing the code instances for the word learning opportunities in text format. The domain expert used these files to review the coding done so far, to develop further the theories involved in the coding, and to adjust the vocabulary accordingly. The detailed analysis PAC facilitated through its coding language was an important part of this development process.

7.2.1.3 Data Analysis using PAC

In the next sections, the data analysis steps performed with PAC are described. These sections are written from the viewpoint of illustrating how PAC was used to facilitate the analysis of the behavioural data. The aim is to establish if and how PAC supported the domain expert in analysing his data. The substantive results of the study are of less importance for the purposes of this thesis.

7.2.1.3.1 Behavioural Input Data

The video recordings of the lessons varied in length from about 15 minutes to 30 minutes. The recordings showed the teacher and the students sitting on the classroom floor, the teacher facing the students and the students forming a half circle. The video taping had been done from a fixed viewpoint. The picture focused well on the group of participants of the lesson without wasting space for irrelevant information. The audio had been recorded with a microphone positioned close to the group on the floor. All participants of the lessons were clearly audible.

Because of the positioning of the participants and the fact that only one static video camera was used for the recording, not every member of the groups could always be seen fully. This caused some problem for the attention coding, which required a frontal view of a student, or for the coding of word learning opportunities, when it was necessary to see to which word on the page the students were pointing.

The interviews with the teachers had a length of about 20 minutes. During the interviews the experimenter and the teacher watched sequences of the video tapes of the lessons and referred to these sequences. The interviews had been transcribed and were available as text files.

Additional data the case study could refer to were copies of the books the groups had read during the lessons and data from the word knowledge tests the domain expert had performed.

For each of the reading books, the domain expert (in conjunction with experienced teachers who were not involved in the data collection) had identified a number of words which were thought to present a challenge to the students. The students were tested regarding their ability to decode and understand these words. Tests were performed before the lessons, immediately after the lessons, and six month after the lessons. The test results were available during the case study but were not incorporated as data into PAC.

Decision on Video Formats

The analog video recordings provided by the domain experts had to be digitised and converted into an appropriate format. Because of the limitations of the JMF video player the maximum possible frame size was half size. To reduce the file size of the video clips, a frame rate of 15 frames per second was chosen. The exact specifications can be taken from Figure 7.2 and an example image from a behaviour recording is given in Figure 7.3.

Format:	Quicktime
Picture compression:	JPEG
Quality factor:	0.5
Frame size:	320 x 240
Frame rate:	15
Audio:	16 bit
Audio compression:	uncompressed

Figure 7.2: Specifications for digital video in Reading Groups case study

7.2.1.3.2 Development of Vocabulary

The development of the vocabulary was based on the different types of coding the domain expert had decided to perform and the analysis planned for the resulting coding sentences. The vocabulary is presented here in its final form as at the end of the case study. It contains the vocabulary for the coding of word learning opportunities, and provision for the coding of procedural and substantive engagement. The substantive engagement had, by the end of this case study, not yet been performed in PAC for reasons described in Sections 7.2.1.1 and 7.2.1.2. Consequently, combination queries of procedural and substantive engagement (which are described in detail in Section 7.2.1.3.4) could not yet be executed. While the actual coding steps for these forms of coding had not yet been performed, a lot of important conceptual work had been undertaken to prepare the vocabulary. For this reason the full vocabulary is displayed here.

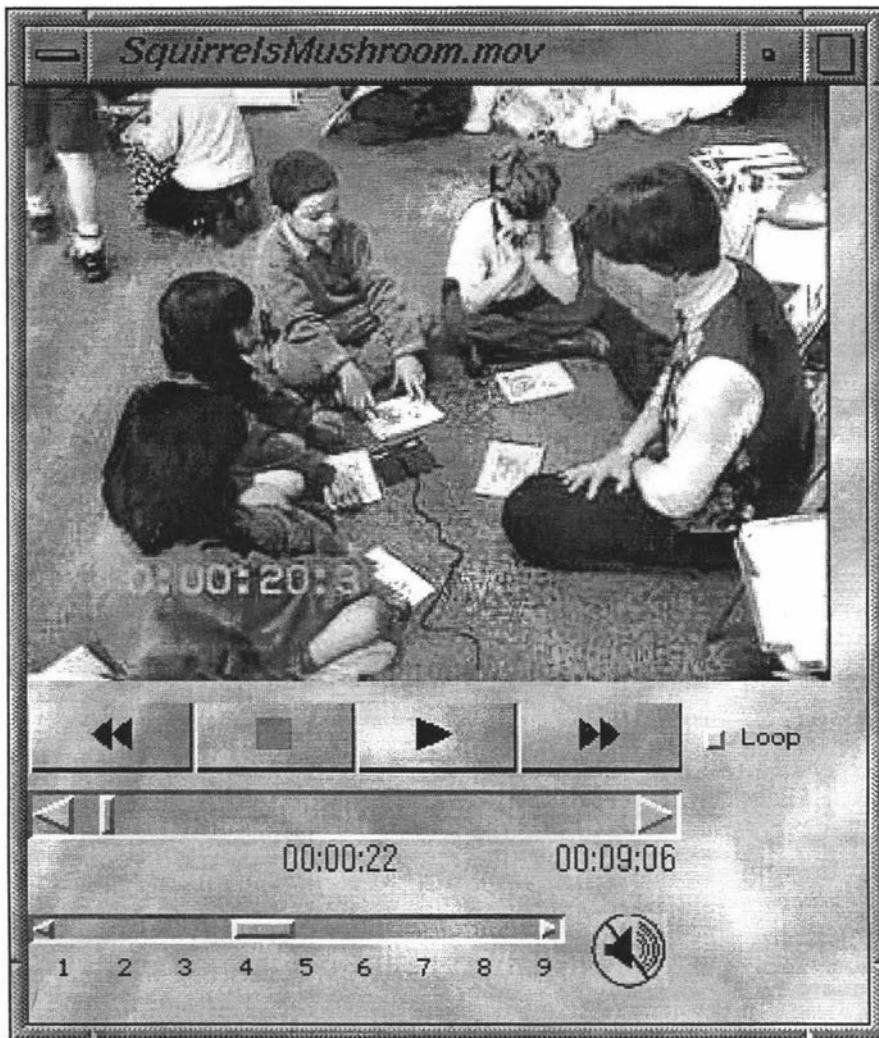


Figure 7.3: Behaviour recording example for Reading Groups case study

The vocabulary needed for the coding of procedural and substantive engagement was easily derived from the specifications given by the domain expert. The vocabulary for the word learning opportunities was based on the preparatory work done in the phase of exploring PAC. The domain expert wrote down a collection of sentences he was likely to use for the description of the word learning opportunities. These sentences were manually checked for their compliance with the grammar of the PAC coding language after identifying the categories of the words involved.

The words of the vocabulary were arranged into hierarchies. This was done for two reasons. Firstly, grouping words into hierarchies meant that this group of words could be handled as one unit in PAC which simplifies the setting up of a study for analysis.

Secondly, building hierarchies facilitated the query process by making it possible to query for the whole group of words within one hierarchy.

The vocabulary used for describing the word learning opportunities evolved over the course of the study. In the process of coding the first few lessons for word learning opportunities, a number of words had to be added or modified. In this phase the domain expert furthered his ideas about the coding and the vocabulary had to be adjusted accordingly. After this initial phase of coding, only a few words had to be added or modified. Some words, which had not been used at all, were removed from the vocabulary. Figure 7.4 lists the essential words of the vocabulary. This listing has been abbreviated by omitting words of the vocabulary which, while necessary for the case study, are of repetitive nature and do not contribute conceptually.

7.2.1.3.3 Coding

According to the study design, three types of coding were to be performed: coding for procedural engagement, coding for substantive engagement, and coding for word learning opportunities.

Procedural engagement was recorded for each target child participating in a lesson in a separate pass through the behaviour recording. Changes in the attention status of a child were recorded initially as point-of-time information and than converted into periods-of-time, covering the whole timeline of the lesson. The procedural engagement data for all target children in a lesson were stored in one description set. In detail, the coding sentences for the procedural engagement contained:

- the name of the child;
- the attention status of the child.

The procedural engagement coding was performed only for few lessons for reasons described in Sections 7.2.1.1 and 7.2.1.2.

Although the substantive engagement coding was not performed in the scope of the case study (see Sections 7.2.1.1 and 7.2.1.2), the format was specified for later integration of the externally conducted coding into PAC. The substantive engagement related to segments of the interview texts. One description set was prepared for the substantive engagement coding for each lesson. The coding sentences were to contain:

Category Person/Thing	disconfirms comments Auxiliaries is are has have	segment2 ... segment12 discussion Text-unit paragraph(s) sentence(s) phrase(s) word(s) syllable(s) letter(s) sound(s)	for of about around before after relatedTo
Teachers Karen Children Mike John Helen Clare Jim ...			
OtherChildren AnotherChild SomeChild Group	Words in Mushrooms RING BEAUTIFUL CLEVER FIND UPHILL	Response toRead toExplain toDefine toSay toReread toListen toPredict toCallOut prior-knowledge meaninG picturE error clue	PositionDesc first second third ReadingDesc oral silent choral ErrorDesc contextAppropriate contextInappropriate gp-similar gp-different ClueDesc letter-sound structure meaning picture
Category Activity	DINNER MYSELF BASKET CLIMB Words in ... Attention Codes Attention PossibleAttention InAttention ExternalInterruption MissingAttentionData AttentionIrrelevant Substantive engagement scale scale 1 scale 2 scale 3 scale 4 scale 5 Pages cover page1 page2 ...		
Activities asks rewardAsks previews explains defines callsOut relates directsAttention reads pointsTo discusses paraphrases makes inserts substitutes omits hesitates ignores pauses rereads selfCorrects says gives tells callsForHelp resolves confirms	Conjs or and while but Preps under on in beside at by with to from	Category Conjunction Conjs or and while but Category Preposition Preps under on in beside at by with to from	TimingDesc immediately waitsThen AttemptDesc successfully unsuccessfully possibly nonverbally not

Figure 7.4: Abbreviated vocabulary in Reading Groups case study

- the name of the child;
- the substantive engagement scale for the child;
- the page number in the book which was read in the lesson.

Word learning opportunities were coded on a point-of-time basis for each lesson, creating a description set for each lesson. The coding focused on word learning opportunities arising in the lessons and described the behaviour of the participants at these opportunities. These descriptions typically contained the following elements:

- the actor, being either one or more target children or the teacher;
- the activity word, often qualified by a descriptor;
- the recipients of the action, being either one or more target children or the teacher;
- the word in the book the word learning opportunity was about;
- the page on the book the word occurred and the count of the occurrence of the word on the page.

Examples of coding sentences for word learning opportunities are displayed in Figure 7.5.

One of the study's requirements was to be able to create references between the different types of coding. The connection between the coding for procedural engagement and for word learning opportunities could be achieved through the common timeline of these descriptions as they were referring to the same behaviour recording.

The connection between substantive engagement and word learning opportunities or between substantive and procedural engagement was more difficult to achieve and required some additional coding. The difficulties would arise because the coding referred to different behaviour recordings, one being a video recording and therefore timebased, and one being a text document without having a timeline. As outlined in the previous section, this problem was solved conceptually but it was not performed in practice before the conclusion of the case study.

The initial idea for linking the data was to use the page numbers in the books read in the lessons as reference points between the different types of coding. This approach proved to be problematic for two reasons. Firstly, it was common for the teacher to read or discuss the same page of a book at different times in the lesson. Therefore the page number could not be the unique identifier of a segment of a lesson. Secondly, to be able to use the page number as reference point, the page number had to be included in the procedural engagement coding for each child. This proved to be inconvenient for the coding process, and had the difficulty that in a number of instances at one point of time children would focus on different pages of the book.

Because the page numbers in the books were not suitable to create the references, it was decided to introduce a segmentation of the lessons. Using the focus of the teacher on a particular page or on a page-independent part of a lesson as guideline, the lessons were

John hesitates before FIND on page3
Helen immediately callsOut word(s) after FIND on page3
Karen waitsThen directsAttention to error on word(s) relatedTo FIND on page3 and oral rereads sentence(s)
Karen tells John toReread sentence(s) with word(s) relatedTo FIND on page3
John oral rereads sentence(s) with word(s) relatedTo FIND on page3 and resolves error on word(s) relatedTo FIND on page3
Karen confirms word(s) relatedTo FIND on page3 to James
Karen oral rereads phrase(s) with word(s) relatedTo FIND on page3
Karen directsAttention to meaning clue for word(s) relatedTo FIND on page3 to John
Mike explains about meaninG of word(s) relatedTo FIND on page3
Karen confirms meaninG of word(s) relatedTo FIND on page3 to John
Karen directsAttention to letter-sound clue for word(s) relatedTo FIND on page3

Figure 7.5: Examples of coding sentences for word learning opportunities in Reading Groups case study

divided into consecutive, non-overlapping segments. Using this segmentation, parts of the interview referring to a particular lesson segment were identified (eg. introduction, segment1, segment2, ..., discussion).

7.2.1.3.4 Querying of Code Instances

The possibilities for extracting information from the code instances had been discussed at length during the design of the case study. The coding scheme had been designed to be able to answer queries such as the following:

- Which word learning opportunities occurred for a particular word?
- What was the attention status of a child when a word learning opportunity occurred?
- How do procedural and substantive coding correlate?
- What sequence of child - teacher interaction occurred around a word learning opportunity?
- What differences in word learning opportunities and child behaviour can be found for words learned and words not learned?

Although the case study design carefully considered elaborate query options, the actual queries performed within the scope of the case study were limited to relatively simple queries applied to the coding of the word learning opportunities. There were two reasons for this limitation:

- The coding for procedural engagement had not yet been done for all lessons. The substantive engagement coding had been performed outside of PAC and had not yet been integrated into the system.
- The restricted access of the domain expert to PAC (see Section 7.2.1.1) made an exploratory querying of the word learning coding difficult.

7.2.1.4 Shortcomings of the Implementation

Quality of Video Replay

Due to limitations of the JMF video player and due to the length of the behaviour recordings and the associated size of the video files, the videos had to be converted to half frame size with a medium quality factor. The domain expert regarded the picture quality of the digital videos as inferior to the quality of the original analog videos.

Although this constituted only a minor problem for coding of the word learning opportunities, it was considered prohibitive for the attention coding. In some instances, the digital video did not provide enough image resolution to convey details of facial expression (such as eye blinking).

Speed of Database Access

The low speed of the database access was annoying. Specific to this case study, in the initial stages, there were frequent changes to the vocabulary and this meant an inconvenient time delay in progressing with the coding. Conceptually, PAC offers great flexibility in working with a changing vocabulary, which is, at least theoretically, mirrored in the implementation. Practically, the slowness of the database access took away part of this flexibility.

7.2.1.5 Extensions and Changes to PAC Caused by the Case Study

The ‘Combined-with’ Query Option

This case study dealt with two forms of behaviour recordings relating to the same sequence of behaviour. The video recordings captured the reading lessons and the stimulated recall interviews, presented in the form of textual transcripts, contained the interviews the domain expert had conducted with the teachers. The interviews referred to segments of the reading lessons in a not necessarily consecutive way, they did not cover the whole lesson, and they sometimes referred more than once to the same lesson segment.

It was agreed with the domain expert that the behaviour of the teacher in the reading lessons could be taken to divide the lessons into consecutive, non-overlapping segments. Sections

of the interviews, referring to a lesson segment, would be marked with the appropriate segment identifier.

This segmentation would form the basis for linking behaviour descriptions for the lessons and for the interviews via their segment identifiers. This segment approach would ensure that behaviour descriptions would be associated with each other that related to the same part of a reading lesson.

To facilitate the segmentation and the linking of descriptions, the ‘combined-with’ query option was introduced. This query option was integrated conceptually into PAC and was implemented in the PAC code. It was tested with some example data but could not be applied in the scope of the case study because of the limitation explained in Section 7.2.1.3.4.

Export of Analysis Results

The initial idea in the development of PAC was that the whole analysis process would be conducted within PAC, resulting in an electronic multimedia presentation of the study results, the conclusion presentation. This idea was not realised for the following reasons:

- The implementation of PAC concerning the interpretation of query or text search results and concerning the conclusion construction was insufficient.
- Only one computer was available to be used for both development work and all of the case studies which limited the time the domain experts could interact directly with PAC.
- In the case of this case study, the domain experts were located in 600km distance and had to travel to use PAC.

For these reasons, the Description, Query and TextSearch modules of PAC were extended to allow the export of data in ASCII format. Code instances were exported as coding sentences text with start and end times. Query results were reported with query text, start and end times, and either a listing of the text string of the individual query results or a count of individual results for the query. Annotations, as results of a text search, could be exported as search text, annotation text, and start and end times.

Time Display

PAC works with a time accuracy of one hundredth of a second. All time values are stored as one hundredth of a second. Originally PAC presented all time values as one number. Dealing with behaviour recordings of up to half an hour in length meant that the display of time values of this resolution was inconvenient for the user. In a number of displays, the time output format was therefore changed to hours:minutes:second.hundredthOfSecond. The internal format for the time values was not modified.

7.2.1.6 Case Study Evaluation

The case study evaluation is based on the case study evidence originating from documents, observations, artefacts, and interviews as outlined in detail in Section 7.1.4.

Usefulness of the coding and query languages, FSCL and FSQL

The coding language allows linking information from different information sources. The construction of the PAC vocabulary forces the analyst to be conceptually more precise. This increased conceptual precision benefited the study. Although it was at times restricting to use a formal language as compared to a natural language, the benefits of the formalised language for querying were understood and appreciated by the domain expert.

The advantages of digital as compared to analog video

Digital video gave the advantage of instant replay of random sequences. The repeated viewing of sequences was possible without effort. According to the domain expert the limitations of the picture quality of the digital video caused by the implementation would have to be overcome.

The possibilities of combining quantitative and qualitative research methods

The potential of combining the quantitative coding for procedural engagement with the qualitative coding for word learning opportunities was thought to be very promising. At this stage the coding of the word learning opportunities was very exploratory. Once theories had emerged, it would be possible to look for a quantitative confirmation of these theories, for example in investigating specific interaction patterns.

The advantages of the combined analysis of behaviour recordings of multiple media formats

A major advantage is to be able to combine the video and text analysis. The domain expert stated that in an educational study, the analysis of video will always be complemented by the analysis of supporting material in text format.

The benefits of a multimedia conclusion construction

For the current case study, the benefits of a multimedia conclusion construction could not be readily seen. The study focused on theory building and the reporting of the study outcomes would have to occur in text format as an academic publication. In a different context, a multimedia conclusion could be a very powerful tool to support the claims of a study.

The possibility of constructing a domain knowledge base

The specific relevance of PAC as compared to other systems for constructing a domain knowledge base could not be established.

To summarise the results of the Reading Group case study, the main research questions can be answered as follows.

Can PAC be used to support the study of behaviour?

PAC can be used to support the study of behaviour. PAC was able to support both the quantitative and qualitative analysis steps which had been defined for the study. Any limitations which were experienced using PAC were due to the implementation and not the underlying concepts. PAC has clearly contributed to gaining a better understanding of the issues involved in the coding for word learning opportunities.

Does PAC provide advanced support for the study of behaviour?

Within the scope of this case study, the underlying study of behaviour could not be completed. Yet even under the reported time and access restrictions, it can be clearly stated that PAC provided advanced support for the study of behaviour. To support this claim the following statements are quoted directly from the interview with the domain expert:

- “The one is the conceptual precision afforded by the opportunity to define the vocabulary for coding, rich descriptions of the data.”
- “Secondly, the opportunity to link information from video with text and hopefully audio and maybe other sources in the future, the opportunity to link that information in a very precise way.”
- “The third major advantage which really goes in with that is the mix of quantitative and qualitative analysis which in future will be very, very valuable.”

7.2.2 The Dog Calming Case Study

This case study was performed in association with a study concerned with looking at the way human behaviour can influence the aggression and submission levels in dogs (McComb, 1999). The behavioural literature describes certain behaviours of dogs or wolves in a pack targeted at modifying the state of aggression or submission of another animal. To calm down an aggressive dog or wolf a behaviour of averting the head would, for example, be displayed. The study referred to here investigates the possibilities of transferring these behaviours into the interaction between humans and dogs. Could it be possible for a person facing an aggressive dog to moderate the dog’s aggression by averting their head?

A number of trials using several dogs, experimenters and interventions were performed and quantitatively analysed. In an experimental setup a dog was put into a cage which allowed the dog some movement but at the same time made it possible to restrict the dog's movement sufficiently to perform the interventions. An experimenter sat directly outside the cage and either stared directly at the dog, constantly blink with their eyes, or faced the dog with their head turned to the side avoiding eye contact. The experiments were captured in video recordings which formed the basis for the subsequent behaviour analysis. Based on observational definitions the dogs' behaviours were scored in time sampling format using paper based score sheets. The observational definitions contained a five scale rating scheme for each of the head, ear, and tail positions of the dogs and for the vocalisations. The five scales distinguished between submissive and aggressive expressions of behaviour. The data collected were analysed according to treatment phase, participating dog, experimenter and observer.

7.2.2.1 Familiarisation with PAC and Scoping of Case Study

The Dog Calming case study was of an exploratory nature. The purpose of the study was to examine how the use of a sophisticated analysis tool like PAC could, compared to traditional analysis tools, facilitate a more accurate, detailed analysis of behaviour. The hypothesis of the domain experts was that a tool like PAC with its digital replay of video and its sophisticated coding language could allow the analysis of behaviour to a level of detail either not possible or simply not feasible with traditional analysis tools.

A number of research questions were formulated. It was agreed that the main emphasis in this case study was to explore new ways to investigate these research questions. It was obvious that it would not be possible in the given time frame to cover all the research questions formulated or to analyse enough data to find well founded answers to the questions posed.

The behavioural literature contains descriptions of aggressive/submissive behaviours of dogs. These descriptions build the basis for classifying the behaviour of dogs as aggressive/submissive. The first research question related to using the detailed description possibilities of PAC to examine the validity of descriptions of dog behaviour as proposed in the literature. The second question was to check the operational definitions used in conducting the dog calming study. In the study operational definitions were used to classify the behaviour of dogs into a five point scale from aggressive to submissive behaviour. PAC was used to perform a detailed, less judgemental description of behaviour to be compared against the scores arrived at by applying the operational definitions. The third question investigated the usefulness of PAC in calibrating observers of behaviour. The fourth question was to look at the correlation of behaviours. It examined if the description

possibilities of PAC allowed a more detailed coding of behaviour which would facilitate the detection of previously unknown correlations.

A number of meetings with the domain experts were conducted to introduce them to the concept and implementation of PAC. After a general introduction to the features of PAC the discussions focused on the possibilities of using digital video and the development of the vocabulary. A few video clips, provided by the domain experts, were digitised and presented in different formats. An initial set of domain specific vocabulary was developed. Following from the definition of the vocabulary, examples for coding sentences, for point-of-time and period-of-time coding, for storing code instances in sets and for querying were presented.

7.2.2.2 Process of Conducting the Case Study

After learning about the concept and implementation of PAC, the domain experts decided on the initial steps in using PAC to support their behaviour analysis tasks. Having defined their study goals they could specify the type of coding necessary to support these goals and the ways in which the resulting code instances were to be queried to extract the information required.

With the context of coding and querying defined the domain experts could setup the vocabulary needed. This vocabulary was strongly based on vocabulary developed during the introductory phase of the study. Parallel to defining the vocabulary the digitised behaviour recordings were introduced into PAC.

In the next step the description sets for the proposed initial coding were defined. For each trial, the coding for four subsets of behaviour, the head, ear, and tail positions and vocalisations, and for external interruptions were conducted.

A range of queries was applied to the resulting code instances. The query results were presented as sequence of coding sentences plus the start and end times of the video sequences relating to the behaviour descriptions. The results were stored as text files and printed.

The domain experts interpreted the query results and defined which additional queries had to be applied to the existing code instances and which additional coding had to be conducted. This led to a cyclic process of coding, querying and interpretation.

7.2.2.3 Data Analysis using PAC

In the next sections, the data analysis steps performed with PAC are described. These sections are written from the viewpoint of illustrating how PAC was used to facilitate the

analysis of the behavioural data. The aim is to establish if and how PAC supported the domain experts in analysing their data. The substantive results of the study are of less importance for the purposes of this thesis.

7.2.2.3.1 Behavioural Input Data

The domain experts selected eleven video clips of 2.5 minutes duration each. Each video clip was a recording of a trial, showing a dog in a cage and the experimenter performing one of the three different interventions. The position of the video camera was static during the whole period of a trial, showing dog and experimenter from one camera angle. The videos contained the audio recording of the trials.

Beside the video recordings, which were the primary behavioural input data, the study referred to the score sheets which had been produced during the conventional analysis of the video recordings in the original study by McComb (1999). The score sheets were presented on paper and were used to identify changes in the scores of the behaviours. The score sheets were not incorporated into PAC.

Decision on Video Formats

The video recordings of the trials were presented by the domain experts as analog recordings. The recordings were digitised and then converted into the appropriate digital format. Because of the limitations of the JMF video player concerning realtime replay of video sequences the video clips had to be reduced to half frame size. Due to the static nature of the scenes recorded and a relatively large area of the video picture bordering the dog and the experimenter the video recording could be cropped before being reduced to half frame size. The full frame rate of 25 frames per second was retained which was possible due to the short length of the video clips. The specifications for the videos are given in Figure 7.6 and an example image is presented in Figure 7.7.

7.2.2.3.2 Development of Vocabulary

The development of the vocabulary was an incremental process. It was closely linked with testing the usefulness and completeness of the vocabulary for coding and querying. To decide on which words were needed as part of the vocabulary the domain experts wrote down the sentences they wanted to formulate to describe the behaviours recorded. The process of writing down these sentences was very useful because it forced the domain experts to think very closely about what precisely they wanted to describe. They discovered many instances where the same behaviour could be described with several different sentences each putting a slightly different emphasis on the description. An example, for instance, are the following sentences: 'dog holding-up head', 'dog has upright head' and 'dog has

Format:	Quicktime
Picture compression:	JPEG
Quality factor:	0.5
Frame size:	320 x 240
Frame rate:	25
Audio:	16 bit
Audio compression:	uncompressed

Figure 7.6: Specifications for digital video in Dog Calming case study



Figure 7.7: Behaviour recording example for Dog Calming case study

upright head-position'. For this study it was agreed to decide on one of the different options for each of the behaviours concerned. The domain experts considered it as not important to

express the nuances in meaning and the later querying was facilitated by having one type of sentence to express each behaviour.

The development of the vocabulary was further influenced by the considerations of the types of coding and querying planned for the study. The decision to provide a continuous description of the vocalisations of the dogs, for example, made it necessary to introduce a way to express that at a given point-of-time no vocalisation occurred. From the different ways the coding language of PAC allows the expression of the non-occurrence of a behaviour, it was chosen to define a specific word which expresses that the dog shows no vocalisation. By grouping the words for vocalisation and non-vocalisation within one hierarchy of words, a PAC query could conveniently extract either sentences relating to both vocalisation and non-vocalisation, only sentences relating to vocalisations or only sentences relating to non-vocalisations.

Because the case study design included comparisons across different treatment phases, interventions, experimenters and observers, words to describe these elements had to be introduced into the vocabulary.

On a few occasions during the study the need arose to add further words to the vocabulary. Due to the flexible design of the coding language this could be done easily. Figure 7.8 shows the vocabulary for the Dog Calming study.

7.2.2.3.3 Coding

To fulfil the objectives of the case study a detailed description of the behaviours displayed in the trials was required. These descriptions were constructed using the coding language of PAC.

The coding sentences contained the following information:

- The phase the trial belonged to: baseline1, baseline2, alternating treatment, preferred treatment1, preferred treatment2;
- the intervention performed by the experimenter: direct stare, head turn, eye blink;
- the experimenter: experimenter1, experimenter2, experimenter3;
- the observer conducting the coding: observer1, observer2, observer3;
- the dog involved in the trial: Max, Diva;
- a description of the behaviour displayed.

Category Person/Thing	RealVocalisation	directStare	Descs(Position)
Dogs	barking	headTum	frontal
Max	growling	eyeBlink	relaxed
Diva	yelping	Phases	raised
Experimenter	whining	baseline1	sideways
experimenter1	General-Action	alternatingTreatments	rear
experimenter2	has	preferredTreatment1	upright
experimenter3	Action	baseline2	lowered
Observer	wagging	preferredTreatment2	turned-away
observer1	turning		
observer2	looking-around	Category Conjunction	Descs(Height)
observer3	looking	and	high
dog		while	medium-high
head			medium-low
ears			low
leg			
tail			unclear
cage		Category Preposition	
ground		Preps	
		between	
		at	
Category Activity			
Vocalisation	Position	Category Descriptor	
	head-position	Pace	
NonVocalisation	tail-position	slowly	
	Intervention	quickly	

Figure 7.8: Vocabulary in Dog Calming case study

In the behaviour descriptions the observers focused on four subsets of the dog's behaviour:

- the position of the head;
- the position of the ears;
- the position or movement of the tail;
- the display of vocalisations.

The observers coded for each of these subsets of behaviour separately. They provided a continuous description for the behaviour of each subset, covering the whole timeline of each video clip with exactly one behaviour description. Figure 7.9 presents examples of coding sentences created.

For each trial two description sets were created. One set contained the code instances relating to vocalisations while the other set contained the code instances for the behaviour descriptions for ears, tail and head. In several trials a rapid change of the vocalisation behaviour occurred. Therefore it was more efficient for the observer to indicate behavioural state changes and use a function of PAC to calculate the period of time for a behaviour associated with a state change. For the coding of the three other behaviours it was decided that the observers would denote the periods of time associated with each behaviour directly.

955	1299	AT DS experl obs1 while Max has lowered nose lowered head-position
1299	5417	AT DS experl obs1 while Max has relaxed head-position
0	561	AT HT experl obs1 while Max has unclear ears
594	1294	B1 DS experl obs1 while Max has raised ears
859	1209	PT1 HT experl obs1 while Max has lowered tail-position and quickly wagging tail
1363	1525	B2 DS experl obs1 while Max has lowered tail-position
9795	9960	B1 DS experl obs1 while Max NonVocalisation
9960	10836	B1 DS experl obs1 while Max whining

(The examples have been taken from several trials. The two numbers give start and end time of the behaviour described in 1/100 of a second.)

Figure 7.9: Examples of coding sentences in the Dog Calming case study

Care was taken so that the periods of time of code instances for each subset of behaviour covered the whole time of a trial but would not overlap.

While the trials were conducted a number of external noises, unrelated to the trials, occurred. For each trial a description set was created which contained descriptions of this external noise if present. Examples for these descriptions are given in Figure 7.10.

2019	2062	B1 DS experl obs1 while noiseDogs
10474	10541	B1 DS experl obs1 while noiseOther
94	255	AT HT experl obs1 while noiseOtherAnimals
14629	14805	AT HT experl obs1 while noiseHuman

(The examples have been taken from several trials. The two numbers give start and end time of the behaviour described in 1/100 of a second.)

Figure 7.10: Examples of external noise descriptions in Dog Calming case study

7.2.2.3.4 Querying and Interpretation

The initial querying of the code instances focused on the four subsets of behaviour within one trial. For each trial, code instances relating to each subset of behaviour were extracted. The resulting code instances were presented in chronological sequence. The relevant FSQL queries are presented in Figure 7.11.

Max ANY-ACTION head-position OR Max ANY-ACTION head
Max ANY-ACTION ears
Max ANY-ACTION tail-position OR Max ANY-ACTION tail
Max Vocalisation

Figure 7.11: Initial queries in Dog Calming case study

Further querying focused on identifying possible correlations between the subsets of behaviours. One subset of behaviour, for example the ear position, was selected and the correlating behaviours, being the tail and head positions and the vocalisations, were identified. The behaviour descriptions were presented graphically for the whole duration of a trial. The graphical representation emphasised sequences of behaviour changes and correlations of behaviours.

In the original study, which worked with score sheets based on operational definitions of behaviour, each subset of behaviour had been assigned a value between one and five, indicating the level of aggression or submission of the dog. For the purpose of the correlation analysis, and aiming at a first step in verifying the operational definitions applied in the original study, the coding sentences produced with PAC were linked to score sheet values. Each coding sentence was assigned a value between one and five and these values were used to group the PAC descriptions. Figure 7.12 gives an example of one of the graphics produced to investigate the correlation of behaviours based on grouped behaviour descriptions. The graphics were developed based on data extracted from PAC but created with a graphics package external to PAC. Beside the information mentioned above the graphic indicates the occurrence of external interruptions during the trial.

7.2.2.4 Shortcomings of the Implementation

Repetitive Setup Tasks

This case study and more so the underlying behavioural study dealt with a relatively large number of behaviour recordings (recordings for different dogs, experimenters, interventions and experimental phases). All behaviour recordings were to be described and the resulting descriptions coded according to the same principles.

In most of its tools PAC requires some initial setup of data. The behaviour recordings have to be introduced as study files, coding, annotation, querying and text search sets have to be defined. While these setup tasks are necessary PAC could offer more convenient interfaces to perform these repetitive tasks. PAC could further offer the option of automated setup performed according to some default settings.

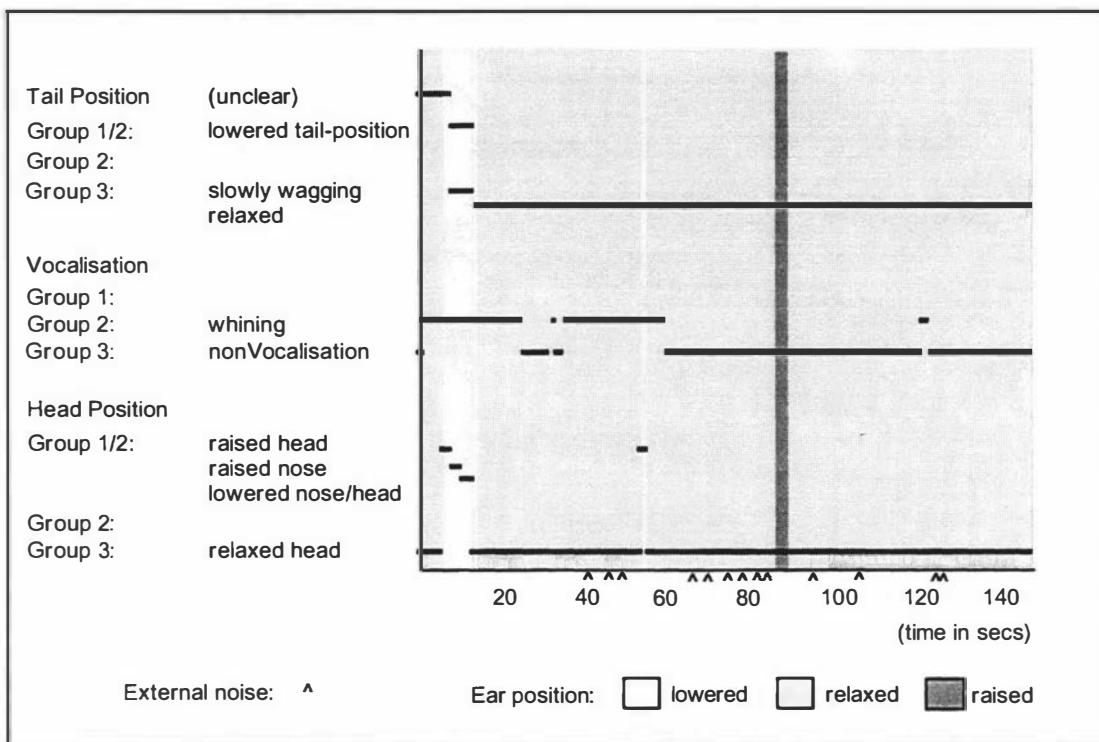


Figure 7.12: Graphical presentation of correlating behaviours in the Dog Calming case study

Storage of Related Query Results

The design of PAC requires the user to define a query set before a query can be issued. The query set defines the range of description sets a query is applied to. While this is necessary for evaluating the query, PAC restricts the user to the storage of only one combination of query text and related results. If the results of a second query are to be saved, the first query data get overwritten. To store the results of two queries, related in terms of referring to the same range of description sets, the user has to define a second query set. While this can be done it makes the storage of query results inconvenient.

Speed of Database Access

The low speed of the database access was annoying. This was felt for example when a change to a different description set was required.

7.2.2.5 Extensions and Changes to PAC Caused by the Case Study

Introduction of a Prefix Component

Most coding sentences in this case study contained a four word component describing the trial parameters and identifying the observer. In the initial implementation such a fixed

component had to be defined via the macros. In the cases where the observers changed from one trial to another (and therefore from one description set to another) while coding for the same set of behaviours this implementation was inconvenient. To change the trial dependent component the observers had to redefine the relevant macros even though the part of the macros describing the behaviour could have remain unchanged.

A prefix component was introduced into PAC. Defined in similar style to a macro, the prefix component was automatically added to the start of a new coding sentence when desired.

Automatic Calculation of Periods of Time

The initial implementation of PAC did not assist the user in calculating periods of time from the times recorded for state changes. If the user coded for state changes a point-of-time value, consisting of a time interval with identical start and end times, was assigned to the code instance.

To increase the flexibility for time based comparisons in the querying of the code instances, while still allowing a convenient recording of state changes, an automated calculation of periods-of-time was introduced. The user would still record the state changes as before the change to the implementation. After recording all state changes the user could request PAC to calculate the periods-of-time from one state to the next state within the description set. These periods-of-time were then stored as new times for the code instances. Should new state changes be inserted or existing state changes be deleted the calculation of the periods-of-time could be performed again. PAC would derive the new periods-of-time by referring to the start times of all code instances.

7.2.2.6 Case Study Evaluation

The case study evaluation is based on the case study evidence originating from documents, observations, artefacts, and interviews as outlined in detail in Section 7.1.4.

Usefulness of the coding and query languages, FSCL and FSQL

The study performed gained from both the vocabulary and the structure of FSCL. Defining the vocabulary prompted the domain experts to critically rethink their operational definitions on a more atomic level. It enabled them to gain a better understanding of what precisely they wanted to measure and made them discover inadequacies in their original descriptions. The structure of the coding language and the fact that there is a language around the vocabulary intensified that process. Using FSCL helped the domain experts to discover that they had “non-logical, non-purpose categories, or overlapping categories”.

The setup of future behavioural studies by the domain experts would likely be influenced by the knowledge of FSCL and FSQL. Operational definitions would be developed less globally earlier on and would be trialed before the full data collection took place with the likelihood of reducing ambiguities and affecting the focus of the study.

FSCL and FSQL provide high potential for studies of social interaction of several individuals, as stated by one of the domain experts: “in a more complex social situation I think the query language would enable us to actually focus on several individuals in a study which would be very, very difficult in the other way”.

The advantages of digital as compared to analog video

Digital video gave the advantage of instant replay of random sequences. The repeated viewing of sequences in different replay speeds was possible without effort and facilitated more accurate coding. The possibility of replaying behaviour sequences from different trials in parallel supports comparison across individuals or interventions.

The possibilities of combining quantitative and qualitative research methods

The domain experts clearly confirmed that PAC supports the combination of quantitative and qualitative research methods through the concepts of the coding and query languages and the various interpretation tools.

Even in the current fairly limited study in which, mainly out of time restrictions, a number of promising analysis ideas could not be followed through, the domain experts see the use of both quantitative and qualitative elements. The study uses a quantitative approach to look for correlations of behaviour, depicting the various behaviours graphically. In a qualitative approach the operational definitions have been re-examined and modified.

The advantages of the combined analysis of behaviour recordings of multiple media formats

In the current study only video recordings of behaviour were analysed. Based on their involvement in other studies the domain experts illustrated the potential for analysing behaviour recordings of multiple media formats. In some studies the combined analysis of video and audio recordings would be of benefit. The pure audio recordings, as compared to video recordings containing both picture and sound information, can be captured far less intrusively and could complement video recordings in sensitive situations. In other studies the video behaviour recordings could be supplemented by information available in textual form. In a study on customs procedures for entering a country after an international flight for example, the customs declaration forms could provide supplementary information for the analysis of behaviour recordings.

The benefits of a multimedia conclusion construction

Already in the current study one very limited form of a multimedia conclusion construction could be used by incorporating visual examples into a printed study report. The construction of an electronic multimedia conclusion as suggested by PAC was seen as having tremendous potential as a teaching and training tool. One of the domain experts had experience in creating illustrative sequences based on several analog video recordings and could recall the enormous effort required in editing the sequences. The use of a tool like PAC would bring a number of advantages. The process of creating the illustrations would be greatly simplified, various media types could be integrated, and additional information in the form of graphical annotations could be added. Beyond the matters of pure presentation, PAC could be used as a computer based learning tool where students or trainees could actively solve tasks like coding of behaviour and compare their results with some model solutions.

The possibility of constructing a domain knowledge base

The domain experts strongly supported the idea of building up a domain knowledge base. They illustrated how each study builds to a certain degree on previous work. Beside containing written reports, as traditionally done, the domain knowledge base could include as well visual examples and detailed information about analysis steps in close relation to the behaviour recordings. This would facilitate efficient building on previous work.

To summarise the results of the Dog Calming case study the main research questions can be answered as follows.

Can PAC be used to support the study of behaviour?

PAC can be used to support the study of behaviour. Looking at the original study by McComb (1999) PAC could have been used to replicate the analysis which had been performed using analog video and conventional paper-based analysis tools. Because no benefit could be seen in replicating an already existing analysis with a different set of tools the case study concentrated on exploring how PAC could be applied for different lines of analysis.

Does PAC provide advanced support for the study of behaviour?

Even in this quite restricted case study PAC did contribute to an advanced analysis compared to using traditional analysis tools. The main strength of PAC becomes apparent in looking towards future studies. PAC is seen as having the potential to strongly benefit the precise and accurate specification of operational definitions. It could aid in the calibration of human observers who might observe behaviour reliable and valid but still not accurate. This would be of enormous benefit to the discipline of behaviour analysis which relies heavily

on human observers. The possibilities of using multimedia conclusions to support teaching and training is seen of great potential. The building of a domain knowledge base over time would strongly help in building on previous knowledge without unnecessarily duplicating work. The domain experts described PAC as a very powerful tool which, as with any superior technology, has to be used wisely. It would be up to the analysts to carefully plan their studies and constantly reflect on the use of the analysis and presentation tools.

7.2.3 The Parkinson's Rating Scheme Case Study

The basis for the third case study was a study concerned with Parkinson's disease (Whittington, 1999). One aspect of this study was to rate the severity of motor disability. As part of achieving this goal, data were collected and analysed according to an existing rating scheme for Parkinson's disease, the Hoehn and Yahr Disease Rating Scale (H&Y).

The study assessed 41 Parkinson's patients according to the H&Y twice with a six monthly interval. The tasks to be performed for the H&Y rating were video taped. The video recordings were analysed using the conventional analysis tools of analog video recorders and paper based observation sheets.

One of the tasks the participants were asked to perform was a finger tapping exercise where the participants attempt to rest their hand on a flat surface and constantly tap with their index finger for a given period of time. The ability to perform this task was assessed separately for each hand and used to rate the severity of motor disability according to a scale value from '0' (less severe) to '4' (more severe).

7.2.3.1 Familiarisation with PAC and Scoping of Case Study

The initial meetings with the domain expert focused on introducing PAC and the Parkinson's disease study. The domain expert was introduced to the concepts and implementation of PAC by highlighting the main ideas behind PAC and by a demonstration of the software. The domain expert provided information about his work on defining an improved rating scheme for Parkinson's disease. First analysis ideas were discussed and tested. An initial vocabulary was developed and some preliminary coding and querying was performed. The domain expert decided on an approach of how to use PAC to add value to his existing analysis approach.

The resulting case study was of exploratory nature. It attempted to indicate ways in which an advanced analysis tool like PAC could be used to achieve a more detailed analysis of video recordings and could lead to a more accurate or faster classification of Parkinson's patients. Similar to the Dog Calming case study (see Section 7.2.2) the emphasis was set on exploring new analysis techniques. It was not expected that the limited amount of data

analysis possible in the given time frame would lead to well-founded results about the Parkinson's disease or a rating scheme.

The case study concentrated on the finger tapping exercises. The standard way of analysing the finger tapping exercises is to watch the recorded video a few times and then to assign the scale value. The general idea of the case study was to investigate if a detailed analysis of these exercises, as possible with an analysis tool like PAC, would allow a more accurate classification of Parkinson's patients than normally achieved. The more detailed analysis concentrated on features like the number of finger taps, frequency, amplitude, hesitations, arrests, and dyskinesias, which are involuntary movements. A subset of video recordings of nine participants was selected for analysis in the case study.

In this case study it was not attempted to achieve an accuracy of quantitative measures as can be reached with specific experimental setups such as touch sensitive pads and electronic measurements. The case study can be distinguished from such a study design as follows:

- The recording of behaviour on video can be achieved in almost any location and is accessible to a wide range of researchers. Video recording can be used to capture finger tapping as well as the other tasks useful in the rating of disease severity. These reasons emphasise the value of video recording for capturing behaviour in context of Parkinson's disease. It is therefore worthwhile to investigate analysis approaches of video recordings of the behaviour.
- The finger tapping of Parkinson's patients contains movements which can only with difficulties be measured electronically. Two examples for such movements would be an 'attempted tap', where a patient would start a tapping movement but is not successful in touching the tapping surface, or an 'interrupted tap', where the tapping movement is interrupted and a small movement into the opposite direction occurs before the tap is completed.

The research questions for the case study were set as follows:

- Investigate correlations between measurements of number of taps, frequency, amplitude, hesitations, arrests, and dyskinesias;
- Try to establish characteristic profiles of the new measures for participants with a certain H&Y rating;
- Compare the H&Y based classification concerning the effect of Parkinson's on the left and right body sides to the results of the new measures;
- Attempt to identify one or two most characteristic behaviours which might be used in future as reliable indicators for classification.

7.2.3.2 Process of Conducting the Case Study

Before the case study could be started the domain expert had to become familiar with the concept and implementation of PAC. Once this had been achieved, scope and objectives of the case study were defined and preparations for the study undertaken. The behaviour recordings were digitised and the vocabulary was developed.

After setting up the description sets the domain expert performed the coding for the different components of the tapping behaviour. This coding was done for the nine participants for each of their left and right hands.

A range of predefined queries was applied to the code instances. The query results were stored in text format and printed. The domain expert used the query printouts to check the code instances for missing data or inconsistencies. Where these were found the data were corrected and the respective queries performed again.

The query results were presented either as counts, as sequences of code instances, or as groupings of code instances. Based on the query outputs the domain expert started to interpret the collected data.

7.2.3.3 Data Analysis using PAC

In the next sections, the data analysis steps performed with PAC are described. These sections are written from the viewpoint of illustrating how PAC was used to facilitate the analysis of the behavioural data. The aim is to establish if and how PAC supported the domain expert in analysing his data. The substantive results of the study are of less importance for the purposes this thesis.

7.2.3.3.1 Behavioural Input Data

The domain experts selected video recordings of the finger tapping task for nine participants. The tapping task had been performed for 30 seconds with each hand. The video recordings focused on the hand performing the tapping task and showed no other part of the participants body. Because both left and right hands were recorded from the same camera perspective the sides from which the hands were seen were different with either thumb and index finger or little finger in the foreground. Due to varying lighting the image quality differed among the recordings.

The recordings contained the tapping sound which was clearly audible for most recordings. For some recordings, specially recordings with very low amplitude tapping, the tapping sound could be heard only with difficulty.

Additional to the behaviour recordings the evaluation of the participants' severity of Parkinson's disease according to the UPDRS, the Hoehn & Yahr Score, and Modified Activities of Daily Living Score were available for reference.

Decision on Video Formats

The domain expert presented the behaviour recordings on analog video tape. The behaviour recordings had to be digitised and converted into an appropriate format.

Because the finger tapping was a relatively fast moving activity (of up to 3 taps per second) it was important to select a video format which would allow realtime replay of video, perfectly coordinating presentation of picture and sound information. Due to the limitations of the JMF video player the frame size of the video had to be reduced to quarter size. With this reduced frame rate the video replay was close to realtime, providing good coordination at least for the slower moving sequences but still visible delays for the faster moving sequences.

The analog video recordings showed the tapping hands relatively well centred in the picture. This allowed cropping a substantial part of the picture before reducing the frame size. With the cropping it was possible to reduce the frame size to quarter size but effectively preserving the relevant parts of the video recordings, the tapping hands, in half frame size. The video formats for this case study are given in Figure 7.13 and an example image from the behaviour recordings is given in Figure 7.14.

Format:	Quicktime
Picture compression:	JPEG
Quality factor:	0.5
Frame size:	160 x 120
Frame rate:	25
Audio:	16 bit
Audio compression:	uncompressed

Figure 7.13: Specifications for digital video in Parkinson's Disease case study

7.2.3.3.2 Development of Vocabulary

The vocabulary for this study was developed in parallel with the development for the analysis ideas for the study. A list of intended coding sentences was compiled and used to identify the words necessary to build these sentences.

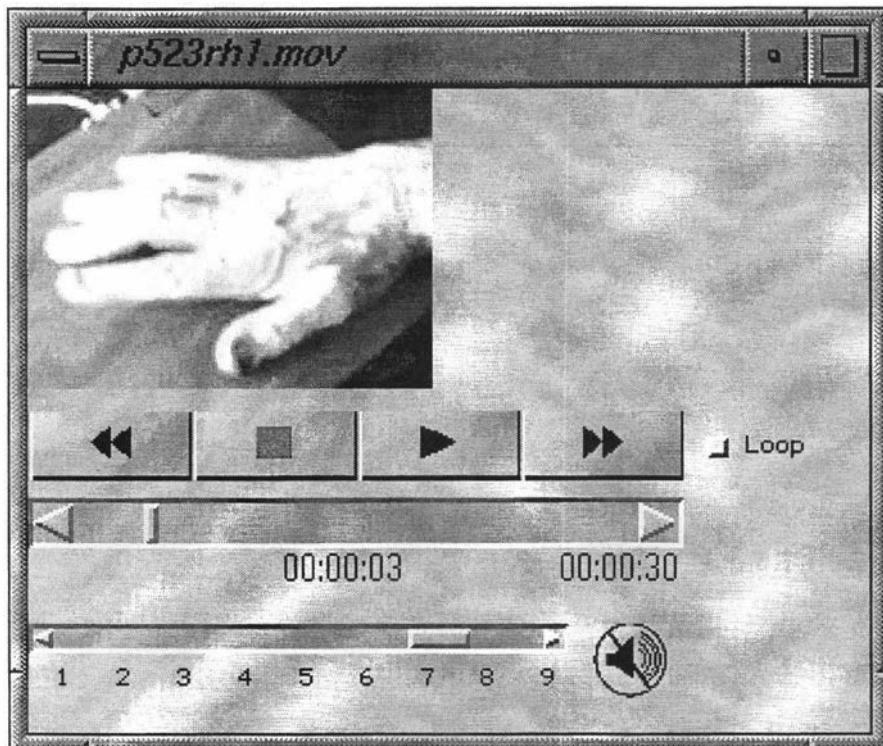


Figure 7.14: Behaviour recording example for Parkinson's Disease case study

Beside the words required to describe the different components of the tapping behaviours words to identify the original study phase, the behaviour recordings originated from, and to name the observers had to be introduced.

Once the details of the study had been finalised the vocabulary remained static. The vocabulary is presented in Figure 7.15.

7.2.3.3.3 Coding

According to the goals defined for the case study the PAC coding language was used to code for the different components of the tapping behaviour for each behaviour recording. These components were:

- occurrence of a tap as point-of-time coding;
- periods-of-time of high/normal/low frequency of tapping to cover the whole timeline;
- periods-of-time of high/normal/low amplitude of tapping to cover the whole timeline;
- periods-of-time of arrest;
- occurrence of tremor, hesitation, dyskinesia of the tapping finger, dyskinesia of the hand as point-of-time.

Category Person/Thing	Observer3	Category Descriptor
Participants	hand	higherLower
p511	H&Y Score	higher
p522	H&Y 1.0	lower
p523	H&Y 2.0	highNormalLow
p524	H&Y 3.0	high
p534	H&Y 4.0	normal
p538	H&Y 5.0	low
p540		leftRight
p541		left
p542		right
p544		
p547		
Observer	Category Activity	Category Conjunction
Observer1	tapping	while
Observer2		Category Preposition
	TappingConcepts	
	frequency	
	amplitude	

Figure 7.15: Vocabulary in Parkinson's Disease case study

Beside the description of the behaviour component each coding sentence contained the following information:

- sequence1 or sequence2 for the first or second behaviour recording for each participant;
- observer1 or observer2 for the observer performing the coding;
- the participant identifier;
- left or right hand.

For each behaviour recording two description sets were defined. One description set was used to store the tap occurrences while all other code instances were stored together in a general set. A number of examples of coding sentences are given in Figure 7.16.

Besides the code instances the general description sets contained annotations. In these annotations any general observations either about the participants behaviour or about the coding process were noted. Examples for these annotations can be studied in Figure 7.17.

The coding for a number of behaviour components posed a problem of judgement. There was no 'mathematical' definition of what constituted high, normal, or low frequency or amplitude, or when a hesitation became an arrest. The judgement of high, normal or low was based on the tapping example the experimenter had given the participants before they performed the tapping task. Guidelines for the other components were provided by the domain expert and were then up to the judgement of the individual observers.

```
sequence1 observer1 while normal amplitude p541 right hand  
sequence1 observer1 while low frequency p538 left hand  
sequence1 observer1 while hesitation p534 left hand  
sequence1 observer1 while dyskinesia(hd) p549 right hand  
sequence1 observer1 while dyskinesia(tp) p534 right hand  
sequence1 observer1 while p541 tapping left hand
```

Figure 7.16: Example coding sentences in Parkinson's Disease case study

Normal tapping with slight loss of control near end.
High frequency and low amplitude combined with tremor/hesitations and some loss of control.
Normal frequency and just normal amplitude with no hesitations but some loss of control including other finger movement.

Figure 7.17: Example annotations in Parkinson's Disease case study

7.2.3.3.4 Querying of Code Instances

A series of queries distinguishing between the participants and their left/right hands were performed. It was queried for descriptions relating to:

- frequency of the tapping;
- amplitude of the tapping;
- number of taps;
- arrest;
- tremor;
- dyskinesia;
- hesitation;
- correlation between amplitude and frequency;
- correlation between hesitation and frequency;
- correlation between hesitation and amplitude;
- correlation between dyskinesia and frequency;
- correlation between dyskinesia and amplitude.

```
frequency p522 left hand
low frequency ANY-PERSON/THING left hand
amplitude p538 right hand
normal amplitude ANY-PERSON/THING right hand
p534 left hand tapping
dyskinesia(hd) p549 right hand
dyskinesia(tp) p534 right hand
hesitation p534 left hand
normal amplitude left hand SAME-AS low frequency left hand WITHIN 0
hesitation left hand SAME-AS normal frequency left hand WITHIN 0
hesitation left hand SAME-AS low amplitude left hand WITHIN 0
dyskinesia(hd) SAME-AS frequency WITHIN 0
dyskinesia(hd) SAME-AS amplitude WITHIN 0
dyskinesia(tp) SAME-AS frequency WITHIN 0
dyskinesia(tp) SAME-AS amplitude WITHIN 0
```

('WITHIN 0' in the 'SAME-AS' queries specifies that only two time periods can be selected which really overlap. A positive value would allow selecting periods which are as many 1/100 seconds apart as specified by the value.)

Figure 7.18: Example queries in Parkinson's Disease case study

A range of example queries is given in Figure 7.18 and some example query results are presented in Figure 7.19. Similar queries were performed for groups of participants. These groups were determined by a rating the parent study had assigned to the participants according to the Hoehn & Yahr score.

The query results were presented either as counts in the case of number of taps per behaviour recording, as sequences of code instances, or as groupings of code instances. Based on the query outputs the domain expert started to interpret the collected data.

7.2.3.4 Shortcomings of the Implementation

Repetitive Setup Tasks

Studies like this case study deal with a large number of behaviour recordings. In most of its tools PAC requires some initial setup of data. The behaviour recordings have to be introduced as study files, coding, annotation, querying and text search sets have to be defined. Since these setup tasks are necessary PAC could offer more convenient interfaces to perform these repetitive tasks. PAC could further offer the option of automated setup performed according to some default settings

Query:	low frequency ANY-PERSON/THING left hand
Result:	2527 3000 sequence1 observer1 while low frequency p538 left hand 1805 3000 sequence1 observer1 while low frequency p541 left hand
Query:	normal amplitude left hand SAME-AS low frequency left hand WITHIN 0
Result:	1916 3000 sequence1 observer1 while normal amplitude p538 left hand SAME-AS sequence1 observer1 while low frequency p538 left hand
Query:	dyskinesia(hd) SAME-AS frequency WITHIN 0
Result:	0 3000 sequence1 observer1 while dyskinesia(hd) p549 right hand SAME-AS sequence1 observer1 while normal frequency p549 right hand

(The ‘SAME-AS’ query option considers only code instances which refer to the same behaviour recording. If two code instances are found which fulfill the query the query result is constructed by combining the two time periods and the two coding sentences.)

Figure 7.19: Example query results in Parkinson’s Disease case study

Realtime Replay of Video

While a bigger image size for the video replay would have been welcomed by the domain expert the biggest criticism was the lack of accurate realtime replay of the behaviour recording. The noticeable discrepancy between the image of a finger tap and the associated tapping sound was disturbing.

Storage of Related Query Results

The design of PAC requires the user to define a query set before a query can be issued. The query set defines the range of description sets a query is applied to. While this is necessary for evaluating the query PAC restricts the user to the storage of only one combination of query text and related results. If the results of a second query are to be saved, the first query data get overwritten. To store the results of two queries, related in terms of applying to the same range of description sets, the user has to define a second query set. While this is possible it makes the storage of query results inconvenient.

7.2.3.5 Extensions and Changes to PAC Caused by the Case Study

Resetting of Macros

In this case study macros were used to conveniently enter, for one description set, the fixed part of each coding sentence which described the study phase and the observer. Initially

only all macros could be reset at once. This was changed to allow the modification of individual macro definitions.

Resetting of Data in the Description Module

Each description set has a whole range of data associated with it. These data are the vocabulary used in the study the set belongs to, the behaviour recordings linked to the set, existing code instances and annotations for the set. If the user chooses to work with a particular description set, these associated data are retrieved from the database and presented in the appropriate format.

Before the user can perform additional coding or can change existing code instances they have to identify the words of the vocabulary associated with the study to be currently used. Further they can define macros to assist with formulating the coding sentences.

In the initial implementation of PAC a change between description sets caused PAC to reset all the data associated with a description set. In case of two sets belonging to the same study and therefore using the same vocabulary this was inconvenient and therefore PAC was modified to keep the setup relating to vocabulary and macros.

7.2.3.5 Case Study Evaluation

The case study evaluation is based on the case study evidence originating from documents, observations, artefacts, and interviews as outlined in detail in Section 7.1.4.

Usefulness of the coding and query languages, FSCL and FSQL

The concepts of the coding and query languages facilitated a structured approach to describing the behaviours. The use of concepts currently used in the research literature was supported. A more detailed description of the behaviours was encouraged.

The advantages of digital as compared to analog video

Digital video gave the advantage of instant replay of random sequences. The repeated viewing of sequences was possible without effort. Slow motion replay at various preset replay speeds facilitated the detailed study of the behaviour recordings.

The possibilities of combining quantitative and qualitative research methods

For this type of study a more precise quantitative analysis could be achieved with different data collection and analysis methods. The use of PAC encouraged the qualitative analysis of the video recordings. PAC would be very effective for the analysis of behaviours which are difficult to quantify like specific walking problems of Parkinson's patients.

The advantages of the combined analysis of behaviour recordings of multiple media formats

(The combined analysis of behaviour recordings of multiple media formats did not occur in this case study and was not commented on in the interview.)

The benefits of a multimedia conclusion construction

(The multimedia conclusion construction was not performed in the case study and was not commented on in the interview.)

The possibility of constructing a domain knowledge base

Construction of a domain knowledge base would be useful for the research area of the domain expert. Neurologists could rate the Parkinson's patients physical symptoms. The behavioural descriptions given by the neurologists could be analysed to assess how symptom severity is determined in actual practice.

To summarise the results of the Parkinson's Rating Scheme case study the main research questions can be answered as follows.

Can PAC be used to support the study of behaviour?

PAC was used in this case study to add a qualitative dimension to the originally quantitative study design. In the interview the domain expert described a potential use of PAC as follows: "The system could be used to build a profile of what constitutes normal and abnormal behaviour if enough data were analysed. This could be compared and contrasted with other methods of quantifying behaviour. This would ensure an accurate description of behaviour being made."

Does PAC provide advanced support for the study of behaviour?

The main advantages of PAC were seen in using digital as compared to analog video and in supporting the qualitative analysis of the behavioural data.

While this case study was conceptualised right from the start as 'exploratory' it was still limited by the relatively small amount of data analysed. The case study was conducted over a time frame of about ten months. A large amount of time was spent on introducing the domain expert to the concepts and implementation of PAC and on discussing and testing various possibilities for coding, retrieval and interpretation. To make a better judgement on the usefulness of the analysis methods explored the behaviour recordings of many more participants had to be analysed.

Additionally it would have been of benefit to the case study if coding schemes for movements beside the finger tapping had been developed. Both of these extensions were not possible because of the time limitations for conducting the case study.

7.3 Cross Case Evaluation

As outlined in Section 7.1.5 the cross case evaluation contains the two elements of looking for replication and diversity. To test for replication the individual case study evaluations were compared on a proposition and main research question basis. Common statements were extracted and are presented in Table 7.2. The comparison of the individual case study results shows that PAC was able to provide advanced support for all three studies of behaviour through its key concepts of the coding and query languages FSCL and FSQL, the support for quantitative and qualitative analysis, and the use of digital video and multimedia technology. Following from these observation it can be stated that the aim of replication was achieved.

The second aspect to be looked at in the cross case evaluation was the multiple case design logic of diversity. The following comparison highlights differences among the individual case studies. It shows the different approaches taken and emphasis set by the case studies and supports the claim that PAC offers a concept suitable for the support of studies of behaviour in a variety of behaviour analysis domains.

- The Parkinson's Rating Scheme study focused on analysing the observable behaviour. The Reading Groups and Dog Calming studies were more interested in the cognitive processes behind the visible and audible information in the recordings.
- All three studies evaluated the visible information while only the Reading Groups and Dog Calming studies interpreted the audible information.
- The Dog Calming and Parkinson's Rating Scheme studies analysed single sources of behavioural data, the video recordings of interventions or exercises. The Reading Groups study worked with two types of behaviour recordings, the lesson and the interview recordings.

7.4 Case Studies Summary

In summary it can be said that conducting the case studies was a very valuable experience and constituted a solid external test of PAC.

The main ideas behind PAC concerning the coding and query languages, the use of digital video, and the combination of quantitative and qualitative analysis methods were confirmed

Case Study Propositions/ Main Research Questions	Cross Case Answers
Usefulness of the coding and query languages, FSCL and FSQL	FSCL and FSQL are useful. They facilitate a more structured and detailed description of behaviour.
The advantages of digital as compared to analog video	Digital video is of advantage compared to analog video with the main advantage being the instant replay of random video sequences.
The possibilities of combining quantitative and qualitative research methods	PAC does support the combination of quantitative and qualitative research methods. This is of advantage for studies of behaviour.
The advantages of the combined analysis of behaviour recordings of multiple media formats	PAC supports the combined analysis of behaviour recordings of multiple media formats. This is of advantage for studies of behaviour.
The benefits of a multimedia conclusion construction	The construction of multimedia conclusions has a high potential to be useful in the context of studies of behaviour.
The possibility of constructing a domain knowledge base	The construction of a domain knowledge base has a high potential to be useful in the context of studies of behaviour.
Can PAC be used to support the study of behaviour?	PAC can be used to support the study of behaviour. PAC was able to cope with given study designs and behaviour recordings.
Does PAC provide advanced support for the study of behaviour?	PAC provides advanced support for the study of behaviour. The benefits of using PAC are based on the coding and query languages, FSCL and FSQL, the support of both quantitative and qualitative analysis methods, and the use of digital video and multimedia technology. The multimedia conclusion construction and the building up of domain knowledge have high potential to be of benefit.

Table 7.2: Answers across case studies

in their usefulness for the study of behaviour. The implementation of PAC proved to be of acceptable standard to be used in the case studies.

The close interaction with the domain experts was very valuable. It provided further insights into their analysis methods and into their requirements for an analysis tool. Besides testing existing concepts in PAC this collaboration generated new ideas to further support the study of behaviour.

Conducting the case studies strongly confirmed the proposition that only such an involvement could put the domain experts into a position to understand and consequently critique PAC. This critique, based on a solid understanding of the concepts behind PAC, seems to be as valuable as the case study results themselves.

Some aspects of PAC have not yet been tested sufficiently. These are mainly the query language and the interpretation and conclusion tools. While the query language has been used and discussed throughout the studies it has not yet been challenged in depth. The

interpretation and conclusion tools provided in the PAC implementation have hardly been used during the case studies. The reasons herefore are the lack of time available for the case studies, deficiencies in the PAC implementation, and the restricted access of the domain aspects to the PAC system to allow for more exploration.

The range and number of case studies performed and the opportunities to intense discussions with a range of researchers from different research backgrounds seems to have provided a sufficient initial test of PAC.

Chapter 8

Conclusions and Further Work

In this chapter the research reported on is reviewed. The contributions of the research are highlighted and further work is identified.

8.1 Review

The research for this thesis can be divided into three phases. In the first phase, the context for the study of behaviour was identified, the relevant literature and existing solutions were reviewed. Based on this work the research goals were formulated. In the second phase, a new conceptualisation for the support of the study of behaviour was suggested. This conceptualisation provided the basis for the specification and implementation of a new application, called PAC. This application PAC was used in the third phase of the research as a tool for conducting case studies in different fields of behaviour analysis. In these case studies the concepts underlying PAC were explored and tested in conjunction with domain experts in the study of behaviour.

Establishment of the Research Task

At the beginning of this research the term ‘study of behaviour’ was introduced to set a wide context for the design and development of a multimedia information system to support the study of behaviour. The study of behaviour occurs in a wide range of disciplines including psychology, education and sociology (Hardyck and Petrinovich, 1975; Rosnow and Rosenthal, 1996; Stangor, 1998) and behaviour is commonly captured in different media formats for later analysis (McBurney, 1994; Heath, 1997; Kent and Foster, 1977; Sommer and Sommer, 1991; Cozby, 1996; and Mars, 1989). Depending on the discipline this analysis focuses on the ‘observable’ aspects of the behaviour, as in areas usually linked to the term ‘behaviour analysis’, or on the cognitive processes behind the behaviour, as in educational research. In areas like usability studies behaviour is analysed in combination with additional data to investigate the performance of a system or instrument. This context for the study of behaviour was established by preliminary contacts with various researchers in the different disciplines.

Arising from this context, the task was to look for the commonalities among these forms of study of behaviour. This was done by investigating the literature on research methodologies

linked to the mentioned disciplines and by analysing existing computer systems for the support of the study of behaviour. Two main common elements were identified:

- Behaviour is recorded for later analysis in video or audio format. These recordings are supplemented by written transcripts and additional data.
- The main analysis steps comprise some form of description of the recorded behaviours, retrieval of data from these description based on some common criteria, and the interpretation of these retrieved data.

Based on these investigations the main research goal was identified as developing a new conceptualisation for an information system to support the study of behaviour captured in multimedia data format. This new information system was to be a generic tool, applicable to most disciplines concerned with the study of behaviour and was to be strongly based in the theories of information systems. This meant more specifically that it had to:

- concentrate on the common analysis elements among the study of behaviour in different disciplines;
- focus on multimedia technology;
- identify and use information systems concepts suitable for the design of an advanced multimedia information system for the support of the study of behaviour.

New Conceptualisation and Implementation of a New Application

A number of areas were identified as central to the research task. The research had to provide advanced support for the coding of the behavioural data and the related query mechanisms. Both quantitative and qualitative analysis methods are applied in the different forms of study of behaviour (Neuman, 1994; Burns, 1997; Krathwohl, 1997; Gay, 1996) and had to be supported. The development of a multimedia system, specially the use of digital as compared to analog video, opened up new possibilities for analysis techniques to be applied to the behaviour recordings.

The Flexible Structured Coding Language, FSCL, and the related Flexible Structured Query Language, FSQL, were designed. FSCL combines the advantages of two approaches for the description of behavioural data. The flexibility and richness of natural language description is preserved and combined with a formalised coding approach to facilitate the correct and complete retrieval of behaviour descriptions. FSCL is based on a flexible vocabulary divided into different categories of words and a grammar defined on these categories. FSQL is built on top of the structure of FSCL. FSQL allows access to the meaning in description sentences formulated with FSCL by answering questions of the style ‘who did what to whom using what’. Additionally, FSQL contains constructs to evaluate Boolean relationships between FSCL sentences and to access position and

sequence properties of the attached multimedia sequences. FSCL and FSQl are designed to support the description and querying of behaviour recordings across multiple media types.

Support for the combined quantitative and qualitative analysis was achieved through a combination of measures. The design of FSCL and FSQl facilitates both analysis styles by allowing natural language like descriptions which can be queried precisely for their meaning. This design, therefore, supports both a rich and flexible qualitative description and a precise, more abstract, quantitative description. The FSCL/FSQl description mechanism is complemented by an annotation facility. Annotations are natural language descriptions which are analysed with a text search mechanism (Weitzman and Miles, 1995). The results of both a FSQl query and a text search can be combined for both quantitative and qualitative interpretation and are referred to as descriptions belonging to one thematic group. The descriptions belonging to one thematic group can be interpreted in quantitative and qualitative style. Descriptive statistics can be calculated for the thematic groups. These statistics can be based on counts, properties of the underlying behaviour recordings and properties of the individual behaviour recording segments attached to the description sentences. The texts of individual description sentences can be studied in close detail or sequences of texts of several description sentences can be analysed. A further form of interpretation is based on the behaviour recording segments attached to the description sentences. These segments can be viewed, studied in detail, analysed with graphical annotations tools, or compared to exemplar behaviour sequences. The design for quantitative and qualitative analysis is complemented by the introduction of the 'set' concept. Behaviour descriptions and thematic groups of descriptions are stored in sets. For each retrieval or interpretation operation, the range of sets the operation is applied to has to be specified. It is possible to apply an operation to just one or to multiple sets. This set mechanism allows the separation of data where necessary, as for the calculation of certain quantitative measures, and the combination of data where of advantage, as in a qualitative comparison of several behaviour recordings.

The use of digital video has distinct advantages compared to analog video. Digital video overcomes the limitations arising from the physical properties of an analog video tape. Instant access to any sequence in a behaviour recording is possible, the recording can be viewed at any speed, and two or more sequences from the same recording can be viewed in parallel. The digital behaviour recordings can be overlayed with graphics elements to provide a new range of analysis techniques. Data of multiple media formats are not only used as input data in the form of behaviour recordings but as well to present the findings of a behavioural study. During the analysis process a range of data in different formats are produced. Behaviour recording sequences are identified which show characteristic or exemplar behaviour. Numeric data in form of tables or graphs are produced. Intermediate

results are captured in text format. All these different elements can be combined at the end of a study to illustrate the overall findings. A multimedia conclusion can be constructed and stored for each study.

After the concepts for a new system to support the study of behaviour had been defined, a prototype application, called PAC, was implemented in the Java programming language. The aim was to create an application which could be used to conduct case studies in collaboration with researcher from different areas of studies of behaviour. The development effort focused on the implementation of the new conceptualisation with special emphasis on the new coding and query languages. Due to technical restrictions the multimedia conclusion construction and the graphical annotation tools could not fully be implemented. Besides enabling the conducting of case studies, implementing the system was a valuable first step in testing the validity of the concepts suggested, a research approach supported, for example, by Nunamaker, Chen and Purdin (1990/91).

Test of the Conceptualisation

Three case studies were conducted using the application PAC. These studies aimed at two closely related outcomes. Firstly, the concepts identified for the support of studies of behaviour were to be evaluated. Secondly, through working with PAC on their studies, the domain experts would become so familiar with the concepts that they were able to comment on the usefulness of these concepts beyond the scope of the implementation and the case studies.

The first case study was concerned with the way children learn to read in small group situations in New Zealand primary schools. PAC was used for the analysis of a subset of the behaviour recordings following the analysis scheme developed for the study independently from PAC. The use of PAC facilitated a conceptually rich and precise description of the cognitive processes displayed in the behaviour recordings. Of specific importance to the study was the possibility for analysis of behaviour recordings across multiple media types.

The second case study investigated how the aggression level of a dog can be modified by the behaviour displayed by a person. In this study the analysis using PAC was added to the original study design. The focus of the study was to investigate how the coding facilities of PAC could be used to achieve a more precise and detailed description of behaviour. The results of the case study indicated that a system like PAC can bring great advantages in areas like observer calibration, observer training and inter-observer reliability.

The third case study was based on the analysis of behaviour recordings of Parkinson's disease patients in the context of the development of an improved rating scheme for the

disease. As in the second case study, an experimental analysis with PAC was added to the original study design. The original coding scheme for the behaviour recordings was supplemented by a much more detailed description of single aspects of the behaviours displayed. While the volume of data analysed using PAC in the case study was limited, it can be assumed that a detailed description as supported by PAC adds valuable information to the original study design.

All three case studies worked extensively with the coding language FSCL. The related query language FSQL was used to perform basic queries. The more advanced features of FSQL, like the Boolean, time and sequences based query options and especially the query option to retrieve information across descriptions relating to multiple media types, were discussed conceptually but not tested sufficiently. Similarly, the interpretation and conclusion construction tools provided by PAC were demonstrated and discussed but not applied in the case studies.

8.2 Contributions

In this research a new conceptualisation to support the study of behaviour was suggested and a prototype application implemented. The usefulness of the conceptualisation was tested and confirmed by conducting cases studies using the implementation. The core contribution of this work is the development of the new flexible structured coding and query languages, FSCL and FSQL. Further contributions are the conceptualisation of a system supporting the combined application of quantitative and qualitative analysis methods, the conceptualisation of a generic system providing support for the study of behaviour across various disciplines, and the conceptualisation of a system providing new ideas for the use of multimedia. In more detail, the contributions can be described as follows:

- *FSCL and FSQL*: In the literature review the main analysis steps performed in the studies of behaviour were identified as description, retrieval and interpretation. The descriptions take either the form of natural language sentences or are based on a formalised coding scheme. The research presented in this thesis introduced a new natural language like coding language, called the Flexible Structured Coding Language, FSCL. FSCL combines the advantages of both description approaches. It provides the flexibility and expressiveness of a natural language approach and at the same time the precision of a formalised coding system. The application of FSCL and of the accompanying query language, the Flexible Structured Query Language, FSQL, in the case studies has confirmed the usefulness of the new coding approach. Specifically, the use of FSCL and FSQL allows a greater conceptual depth in the description of behaviour and of the cognitive processes behind the behaviour, it supports the analysis

of behaviour recordings across different media types, and the combination of quantitative and qualitative analysis methods.

- *Quantitative and qualitative analysis methods:* Studies of behaviour are conducted applying either quantitative or qualitative analysis methods, largely depending on the disciplines the studies originate from. The literature review shows strong support for the combination of both analysis methods within one study (Howe, 1988; Brannen, 1992; Hammersley, 1992; Bryman, 1988; Bryman, 1992; Greene, 1989; Patton, 1990; Miles and Huberman, 1994; Reichardt and Rallis, 1994; Dey, 1993). The research presented here highlights the common analysis steps of description, retrieval, and interpretation, performed in both methods, and then introduces several mechanisms to support the combination of both methods. These mechanisms are the new coding and query languages, the use of sets to group or separate analysis results, and the provision of quantitative and qualitative interpretation tools within one application. The ability of the conceptualisation to support both quantitative and qualitative analysis methods has been confirmed by the domain experts involved in the case studies.
- *Cross disciplinary approach:* The study of behaviour is performed in a range of different disciplines and existing applications originate from these different disciplines. Not much literature exists indicating efforts across the boundaries of the disciplines. The research presented here looks across the disciplines at the various efforts from an information systems perspective and attempts to identify the common elements of the study of behaviour captured on multimedia data. This approach brings two advantages. Firstly, looking across disciplines allows the collection of useful tools and analysis techniques from the various disciplines and implementations. This can lead to the definition of a potentially richer system and to the support across quantitative and qualitative analysis methods. Secondly, the information systems perspective introduces elements like structured design, databases, formal languages, or user interface issues into the design of a system.
- *Multimedia system:* While a number of existing systems to support the study of behaviour have grown from text-based systems, the concepts introduced in this research concentrate on the design of a multimedia system. This approach has the advantage of being able to include elements relating to multimedia data right from the start. The coding and query languages are designed to support the analysis across different media types. Some analysis tools are suggested which replicate the use of 'pen and paper' on multimedia data. Data collected and produced during the analysis process can be combined into a multimedia conclusion presentation.

The concepts introduced in this research have been implemented in a prototype application, called PAC. This application has made it possible to test the concepts introduced by performing case studies in collaboration with domain experts. Working in these case studies with the application PAC has allowed the domain experts to familiarise themselves with the concepts behind PAC and therefore to be able to give well-founded feedback on these concepts. The implementation of PAC has therefore been an important element in the research process.

8.3 Further Work

A number of topics for further research have arisen from the work on this thesis. These topics can be grouped under the headings coding and query languages, multimedia analysis, implementation PAC, and extension of scope and interaction with domain experts. The further research topics relate both to the extended testing of ideas suggested in this work and to the development of further conceptualisations.

The Coding and Query Languages

The Flexible Structured Coding Language, FSCL, and the Flexible Structured Query Language, FSQL, have both been proven to be very useful to support the study of behaviour. A number of extensions based on the current work are possible:

- Further studies have to be conducted to fully test all features of both FSCL and FSQL. Two interesting areas are the analysis across different media types and the combination of quantitative and qualitative interpretation based on one set of coding sentences.
- FSQL can currently only match a query sentence with a coding sentence when both sentences have the same structure. Yet two sentences can have different structures and still represent the same content. An example for this are the two sentences ‘Bill plays and laughs’ and ‘Bill plays and Bill laughs’. While both sentences describe that a person ‘Bill’ is performing two activities, playing and laughing, the sentence structures are different, as the second sentence explicitly states that ‘Bill’ is the actor performing the activity laughing. FSQL could be extended to allow matching of two sentences like the example sentences presented here which share meaning but not structure.
- FSCL and FSQL have been designed to support the study of behaviour. The languages are used to describe and analyse behaviour captured in form of multimedia recordings. It would be interesting to investigate how the concepts of FSCL and FSQL could be used in the areas of knowledge based systems or artificial intelligence. In these areas knowledge needs to be represented and an interface for querying this knowledge base has to be implemented. The approach of FSCL and FSQL, to provide a natural language

like description language in which description sentences can be queried for their meaning, could be valuable for these areas. It had to be investigated how the grammars of FSCL and FSQL could be extended to allow the formulation of questions as clausal variations.

- Universities, like Massey University, have an increasing number of international students. Not all of these students have sufficient proficiency in the teaching language English and might therefore have difficulties to understand and respond to teaching material presented in computer-based learning systems. The use of FSCL to describe the teaching material and to provide an automated translation of material and questions into natural languages other than English could be investigated. To do this the sentence structure of these other languages would have to be analysed, with appropriate categories and grammars developed. The different words of the natural language and the FSCL categories and grammars would have to be mapped from one language into the other.

Multimedia Analysis

The focus of this research has been on providing a multimedia analysis system to support the study of behaviour. A number of extensions concerning the multimedia aspect are possible:

- The term ‘study of behaviour’ had been defined to include a wide range of analysis foci from the behaviours displayed to the study of the cognitive processes motivating the behaviours. This interdisciplinary perspective for constructing a multimedia system could be widened to provide better analysis support for other areas. In usability studies, for example, it is essential to integrate measurements about the systems studied into the analysis of the behaviour recordings. It might be possible to perform some form of automated coding for these measurements, using predefined FSCL sentences, to support their integration with the analysis of manual coding done on the behaviour recordings.
- The current research suggests some tools to support the analysis of video pictures and to transfer analysis techniques used with pencil and paper into a multimedia system. The implementation PAC only provides very basic tools to display the possibilities of graphical annotation of video recordings to the domain experts. Besides improving the implementation, there is space for conceptual extension. Graphical annotations could be treated as a third form of description beside coding and textual annotation. The annotation colours or shapes could be used to retrieve data based on graphical annotations. It might be possible to start with primary colours for initial description and

then to merge these colours according to rules for colour composition to visually represent the overlapping of descriptions.

The Implementation PAC

While the implementation PAC was of sufficient quality to demonstrate the main concepts of the research and to conduct the case studies it leaves much space for improvement:

- Looking at user interface issues a better interface to FSCL and FSQSL is needed. The vocabulary for FSCL can easily contain one hundred or more words to be used in one study. These words have to be easily accessible by the user while at the same time leaving enough screen area to present the behaviour recordings and newly constructed coding sentences. The current interface to the vocabulary of FSCL is cumbersome and takes up a large proportion of the screen. An improved interface could be built by using voice recognition techniques. With the, compared to natural language, small vocabulary, voice recognition based input of coding sentences should be fairly accurate. The advantages would be that less screen space had to be reserved for displaying the vocabulary and the input of the coding sentences would be more convenient than the current point and click mechanism.
- PAC uses, in contrast to most existing applications in the field of study of behaviour, a database management system to store its analysis data. The current database used is a relational database. While the relational approach is appropriate for storing a number of the PAC data an object oriented database might be more suitable for storing coding sentences. Within PAC the FSCL coding sentences are handled as objects, called semantic trees. Reading and writing these semantic trees from and to the database, the objects currently have to be converted from relational structures into objects and vice versa. A similar problem with using a relational database occurs in storing and accessing the vocabulary, which consists of hierarchically structured word groups.
- The concept behind PAC proposes a multimedia conclusion construction. Due to technical restrictions this multimedia conclusion construction has not been fully implemented in PAC and could therefore not be tested in the case studies. A new implementation has to implement the conclusion construction as suggested in the conceptualisation, using audio to record the conclusion statements.

Extension of the Scope

The application PAC has so far been used in collaboration with a small group of researchers to conduct studies of behaviour. The concept of PAC includes the construction of a multimedia conclusion to represent the findings of a study and the collection of several

study data and results to, over time, build a knowledge base. Further research could investigate how these results could be utilised in a wider context:

- Teaching systems could be constructed where the students could study the results produced with PAC or could follow the analysis steps taken in an exemplar study. From simple browsing this could be extended to a system where the students would undertake their own analysis steps which could then be compared against steps stored already in the system. University lecturers or teachers could use PAC to conveniently construct example material for studies of behaviour for their classes. PAC could be used
- to create annotated archives of video or audio material. Collections of television productions, for example, could be described using the coding language of PAC. These descriptions could, at a later point of time, be searched for queries not anticipated at time of description.
- Following on from the ideas about the use of PAC for various purposes a two level system could be implemented. The full analysis system would allow access to all tools within PAC to perform studies of behaviour. A restricted ‘browsing’ system could be developed to be used only for looking at already performed studies. This system could be based on the world wide web to allow for easy access and extended availability. To implement such a browsing system a range of issues from technical, such as conversion of PAC data structures into Html code, to ethical, such as access restrictions for personal sensitive data, had to be considered.

Interaction with Domain Experts

During the course of this research domain experts from various areas of the study of behaviour have been consulted. To familiarise these experts with the concepts of PAC numerous meetings were conducted and documents exchanged. Case studies were performed, based on studies these experts were currently undertaking, to clarify the concepts of PAC. While this approach was successful, it was as well very time consuming and intensive. Further research could investigate other approaches to introduce new concepts to the user community with the goal of enabling this community to give well-founded feedback on the applicability of the new concepts to their domains.

References

- Agnew, N.Mck. and Pyke, S.W. (1978). *The Science Game: An Introduction to Research in the Behavioral Sciences*. Englewood Cliffs: Prentice-Hall.
- Akeroyd, A.V. (1991). Personal Information and Qualitative Research Data: Some Practical and Ethical Problems arising from Data Protection Legislation. In N.G. Fielding and R.M. Lee (Eds.), *Using Computers in Qualitative Research*, London: Sage Publications.
- Arnold, M., Kuk, G. and Ritter, F.E. (1995). Software Review MacShapa. *Psychology Software News*, 6(1).
- ATLAS.ti (1999). ATLAS.ti: The Knowledge Workbench. <http://www.atlasti.de>, accessed 26/05/1999.
- Avison, D.E. and Nandhakumar, J. (1995). The Discipline of Information Systems: Let Many Flowers Bloom! In E.D. Falkenberg, W. Hesse and A. Olive (Eds.), *Information System Concepts, Towards a Consolidation of Views*. London: Chapman & Hall.
- Banister, P. (1994). Report Writing. In P. Banister, E. Burman, I. Parker, M. Taylor and C. Tindall (Eds.), *Qualitative Methods in Psychology, A Research Guide*. Buckingham: Open University Press.
- Banister, P., Burman, E., Parker, I., Taylor, M. and Tindall, C. (1994). *Qualitative Methods in Psychology, A Research Guide*. Buckingham: Open University Press.
- Bentzen, W.R. (1997). *Seeing Young Children: A Guide to Observing and Recording*. Albany, New York: Delmar Publishers.
- Bordens, K.S. and Abbott, B.B. (1996). *Research Designs and Methods, A process Approach*. Mountain View, California: Mayfield Publishing Company.
- Brannen, J. (1992). Combining qualitative and quantitative approaches: an overview. In J. Brannen (Ed.), *Mixing methods: qualitative and quantitative research*, Avebury: Ashgate Publishing Company.
- Bryman, A. (1988). *Quantity and Quality in Social Research*. vol. 18, London: Unwin Hyman.
- Bryman, A. (1992). Quantitative and qualitative research: further reflections on their integration, In J. Brannen (Ed.), *Mixing methods: qualitative and quantitative research*. Avebury: Ashgate Publishing Company.
- Bryman, A. and Cramer, D. (1994). *Quantitative Data Analysis for Social Scientists*. London: Routledge.
- Burns, R.B. (1997). *Introduction to Research Methods*. Melbourne: Addison Wesley Longman.

- Carter, P.D. (1997). *An Information System for Psychotherapy Research using Expert Commentary of Videotaped Expertise*. PhD thesis, Massey University, New Zealand.
- Carter, P. and Patrick, J. (1997). *An Information System for Qualitative Data Analysis: Flexible Environment for Research And Learning (FERAL)*. Technical Report 9/97, Department of Information Systems, Massey University, New Zealand.
- Carter, P., Patrick, J. and Deane, F. (1998). Managing and analysing multiple expert commentary of videotaped psychodrama sessions using FERAL. *Conference Proceedings Computers in Psychology*. CTI Centre for Psychology, University of York, York.
- Cartwright, A. (1998a). Code-A-Text. *Conference Proceedings Computers in Psychology*. CTI Centre for Psychology, University of York, York.
- Cartwright, A. (1998b). *Code-A-Text*. <http://www.codeatext.u-net.com/ctdetail.htm>, accessed 08/06/1998.
- Cozby, P.C. (1997). *Methods in Behavioral Research*. Mountain View, California: Mayfield Publishing Company.
- Darst, P.W., Zakrajsek, D.B. and Mancini, V.H. (1989). Systematic Observation: An Introduction. In *Analysing Physical Education and Sport Instruction*. Human Kinetics Books, Champaign, Illinois.
- Date, C.J. (1995). *An Introduction to Database Systems*. Addison-Wesley Publishing Company.
- Denzin, N.K. and Lincoln, Y.S. (1998). Introduction: Entering the Field of Qualitative Research. In N.K. Denzin and Y.S. Lincoln (Eds.), *The landscape of qualitative research: Theories and issues*. Thousand Oaks, California: SAGE Publications.
- Dey, I. (1993). *Qualitative data analysis: a user-friendly guide for social scientists*. London: Routledge.
- Dowrick, P.W. (1991). *A Practical Guide to Using Video in the Behavioral Sciences*. John Wiley & Sons.
- Dunham, P.J. (1988). *Research Methods in Psychology*. New York: Harper & Row Publishers.
- Durkin, T. (1997). Using Computers in Strategic Qualitative Research. In G. Miller and R. Dingwall (Eds.), *Context & Method in Qualitative Research*. London: Sage Publications.
- Fassnacht, G. (1982). *Theory and Practise of Observing Behaviour*. New York: Academic Press.
- Fisher, M.D. (1994). *Applications in Computing for Social Anthropologists*. London: Routledge.

- Franks, I.M., Johnson, R.B. and Sinclair, G.D. (1988). The Development of a Computerized Coaching Analysis System for Recording Behavior. In *Sporting Environments, Journal of Teaching in Physical Education*. Vol8, pp. 23-32.
- Fraser, D. (1999). *QSR NUD*IST Vivo, Reference Guide*. Melbourne: Qualitative Solutions and Research Pty.
- Freeborn, D., (1995). *A Course Book in English Grammar*. London: The Macmillan Press Ltd.
- Galliers, R.D. (1994). Choosing Information Systems Research Approaches. In R.D. Galliers (Ed.), *Information Systems Research: Issues, Methods and Practical Guidelines*. Henley-on-Thames: Alfred Waller Ltd.
- Gambrill, E.D. (1977). *Behavior Modification: Handbook of Assessment, Intervention, and Evaluation*. San Francisco: Jossey-Bass Publishers.
- Gay, L.R. (1996). *Educational Research: Competencies for Analysis and Application*. New Jersey Prentice-Hall.
- Goldman-Segall, R. (1993). Interpreting video data. In *Journal for Educational Multimedia and Hypermedia*. Association for the Advancement of Computing in Education, 2,3, pp. 261-282.
- Goldman-Segall, R. (1995). Configurational Validity: A Proposal for Analyzing Ethnographic Multimedia Narratives. In *Journal of Educational Multimedia and Hypermedia*. 4(2/3), 163-182.
- Goldman-Segall, R., Marcovici, M. and Halff, L. (1993). Constellations 2.0: Multimedia Software for Digital Video and Text Analysis. *MERLin, the Multimedia Ethnographic Research Lab*. Faculty of Education, University of British Columbia.
- Gottman, J.M. and Roy, A.K. (1990). *Sequential Analysis: A Guide for Behavioral Researchers*. Cambridge, UK: Cambridge University Press.
- Greene, J.C., Caracelli, V.J. and Graham, W.F. (1989). Toward a Conceptual Framework for Mixed-Method Evaluation Designs. In *Journal of Educational Evaluation and Policy Analysis*. 11(2).
- Hammersley, M. (1992). Deconstructing the qualitative-quantitative divide. In J. Brannen (Ed.), *Mixing methods: qualitative and quantitative research*. Avebury: Ashgate Publishing Company.
- Hammersley, M. (1996). The relationship between qualitative and quantitative research: paradigm loyalty versus methodological eclecticism. In J.T.E. Richardson (Ed.), *Handbook of Qualitative Research Methods*. Leicester: The British Psychological Society.
- Hardyck, C. and Petrinovich, L.F. (1975). *Understanding Research in the Social Sciences: A Practical Guide to Understanding Social and Behavioral Research*. Philadelphia: W.B. Saunders Company.

- Harrison, B.L. and Baecker, R.M. (1992). Designing Video Annotations and Analysis Systems. *Vision Interface 92 Proceedings*. Vancouver, pp. 157-166.
- Harrison, B.L. and Chignell, M.H. (1998). Video Exploratory Sequential Data Analysis (VESDA), VANNA. <http://anarch.ie.utoronto.ca/publications/vesda.html/van.html>, accessed 17/06/98.
- Hartmann, D.P. and Wood, D.D. (1990). Observational Methods. In, A.S. Bellack, M. Hersen and A.E. Kazdin (Eds), *International Handbook of Behavior Modification and Therapy*. New York and London: Plenum Press.
- Heath, C. (1997). The Analysis of Activities in Face to Face Interaction Using Video. In D. Silverman (Ed.), *Qualitative Research, Theory, Method and Practice*. London: SAGE Publications.
- Henderson, D.A.Jr. and Card, S.K. (1986). Rooms: The Use of Multiple Virtual Workspaces to Reduce Space Contention in a Windows-Based Graphical User Interface. *ACM Transactions on Graphics*. Vol.5, No.3, Pages 211-243.
- Henwood, K.L. (1996). Qualitative inquiry: perspective, methods and psychology. In J.T.E Richardson (Ed.), *Handbook of Qualitative Research Methods*. Leicester: The British Psychological Society.
- Hesse-Biber, S., Dupuis, P. and Kinder, T.S. (1991). HYPEResearch: A Computer Program for the Analysis of Qualitative Data with an Emphasis on Multimedia Analysis and Hypothesis Testing. *Social Science Computer Review*. Vol.9, pp. 452-460.
- Hoiem, D.E. and Sullivan, K.D. (1994). Designing and using integrated data collection and analysis tools: challenges and considerations. *Behaviour & Information Technology*. Vol13, Nos 1 and 2, pp. 160-170.
- Hoogland, A.J. (1995). CAMERA: a system for behavioural observation from video recordings. *Psychology Software News*, 5(2).
- Howe, K.R. (1988). Against the Quantitative-Qualitative Incompatibility Thesis or Dogmas Die Hard. *Educational Researcher*. 17(8), pp. 10-16.
- Herschel, R. (1974). *Einfuehrung in die Theorie der Automaten, Sprachen und Algorithmen*. Muenchen Wien: R. Oldenbourg Verlag.
- Huberman, A.M. and Miles, M.B. (1994). Data Management and Analysis Methods. In N.K. Denzin and Y.S. Lincoln (Eds.), *Handbook of Qualitative Research*. Thousand Oaks: Sage Publications.
- Jackson, H., (1980). Analyzing English: *An Introduction to Descriptive Linguistics*. Oxford: Pergamon Press Ltd.
- Keene, J. (1994). Software Review The Observer. *Behavioural & Cognitive Psychotherapy*. Vol22, pp. 181-183.

- Kelle, U. (1995). Introduction: An Overview of Computer-aided Methods in Qualitative Research. In U. Kelle (Ed.), *Computer-Aided Qualitative Data Analysis*. London: Sage Publications.
- Kent, R.N. and Foster, S.L. (1977). Direct Observational Procedures: Methodological Issues in Naturalistic Settings. In, Eds. A.R Ciminero, K.S. Calhoun, and H.E Adams (Eds.), *Handbook of Behavioral Assessment*. New York: JohnWiley & Sons.
- Krathwohl, D.R. (1998). *Methods of Educational & Social Science Research: An Integrated Approach*. New York: Addison Wesley Longman.
- Krauss, R.M., Morrel-Samuels, P. and Hochberg, J. (1988). VIDEOLOGGER: A computerized multichannel event recorder for analysing videotapes. *Behavior Research Methods, Instruments, & Computers*. 20 (1), pp. 37-40.
- Lazarus, J. (1994). Software Review The Observer, version 3.0. *Psychology Software News*. York.
- Lee, R.M. and Fielding, N.G. (1991). Computing for Qualitative Research: Options, Problems and Potential. In N.G. Fielding and R.M. Lee (Eds.), *Using Computers in Qualitative Research*. London: Sage Publications.
- Leech, G., Deuchar, M. and Hoogenraad, R. (1982). *English Grammar for Today: A New Introduction*. London: The Macmillan Press Ltd.
- Lennon, J.A. (1997). Hypermedia Systems and Applications: World Wide Web and Beyond. Berlin: Springer Verlag.
- Louden, K.C. (1997). *Compiler Construction: Principle and Practise*. Boston: PWS Publishing Company.
- Mackay, W.E. (1989). EVA: An experimental video annotator for symbolic analysis of video data. *SIGCHI Bulletin*, Vol.21, No.2.
- Mackay, W.E. and Davenport, G. (1989). Virtual Video Editing in Interactive Multimedia Applications. *Communications of the ACM*, 32(7).
- Mars, H., van der (1989). Systematic Observation: An Introduction. In P.W. Darst, D.B. Zakrajsek and V.H. Mancini (Eds.), *Analysing Physical Education and Sport Instruction*. Champaign, Illinois: Human Kinetics Books.
- Martin P. and Bateson, P. (1993). *Measuring Behaviour: An Introductory Guide*. Cambridge: Cambridge University Press.
- Martin, G. and Pear J. (1992). *Behavior Modification: What it is and how to do it*. New Jersey: Prentice-Hall International.
- McBurney, D.H. (1994). *Research Methods*. Pacific Grove, California: Brooks/Cole Publishing Company.
- McComb, S. (1999). *Human Calming of Dog Arousal*. Manuscript in preparation.

McGeorge, D.W.F. (1987). Video and Occupational Therapy: the Use of Video Recording with a Camera in Therapeutic Settings. *British Journal of Occupational Therapy*, 50(1).

McKenzie, T.L. and Carlson, B.R. (1989). Systematic Observation and Computer Technology. In P.W. Darst, D.B. Zakrajsek and V.H. Mancini (Eds.), *Analysing Physical Education and Sport Instruction*. Champaign, Illinois: Human Kinetics Books.

McNeill, P. (1985). *Research Methods*. London: Tavistock Publications Ltd.

MERLin (1998). <http://www.merlin.ubc.ca/tools/C25/index.html>, accessed 25/06/1998.

Merriam , S.B. (1998). *Qualitative Research and Case Study Applications in Education*. San Francisco: Jossey-Bass Publishers.

Miles, M.B. and Huberman, A.M. (1994). *Qualitative Data Analysis*. London: Sage Publications.

Monk, A.F. and Watts, L. (1995). A poor quality video link affects speech but not gaze. *Proceedings CHI'95*, Conference on Human Factors in Computing, Denver, Colorado.

Muhr, T. (1991). ATLAS/ti - A Prototype for the Support of Text Interpretation. *Qualitative Sociology*. 14(4).

Neuman, W.L. (1994). *Social Research Methods: Qualitative and Quantitative Approaches*. Boston, Massachusetts: Allyn and Bacon.

Noldus Information Technology (1998a). Base Package for Windows, Version 3.0. Specifiacions The Observer.

Noldus Information Technology (1998b). The Observer Video-Pro, Version 4.0 for Windows. Specifications The Observer.

Noldus, L.P.J.J. (1991). The Observer: A software system for collection and analysis of observational data. *Behavior Research Methods, Instruments, & Computers*, 23(3), pp. 415-429.

Nunamaker, J.F.Jr, Chen, M. and Purdin, T.D.M. (1990/91). Systems Development in Information Systems Research. *Journal of Management Information Systems*, Vol 7, No 3, pp 89-106.

Parker, C.M., Wafula, E.N., Swatman, P.M.C. and Swatman, P.A. (1994). Information Systems Research Methods: The Technology Transfer Problem. *Proceedings Fifth Australasian Conference on Information Systems*. Monash University, Victoria. pp. 197-208.

Patrick, J.D. (1985a). The CABER System. *Australian Computer Science Communications*, 7(1), pp.10.1-10.12.

Patrick, J.D. (1985b). The CABER Project: The Capture and Analysis of Behavioural Events in Realtime. *Proceedings ACM Annual Conference*, 92-98, Denver.

- Patrick, J.D. and Chong, K.E. (1987). Real-time inductive inference for analysing human behaviour. *Proceedings Australian Joint Artificial Intelligence Conference*, pp. 305-322, Sydney.
- Patrick, J.D., Ho, L.T. and John, E. (1985). A Generator of LR(1) Parsers for real-time Capture of Narratives. *Australian Computer Science Communications*, 7(1), pp. 11.1-11.11.
- Patrick, J.D. and Lowdon, B. (1986). Computer controlled video replay of player activity in sport. *Sports Coach*, 10(3), pp. 20-2.
- Patrick, J.D. and McKenna, M.J. (1986a). A generalised computer system for sports analysis. *Aust. Journal of Science and Medicine in Sport*, 18(3) pp. 19-23.
- Patrick, J.D. and McKenna, M.J. (1986b). Computerized analysis of handball in Australian Rules Football. *Australian Journal of Science and Medicine in Sport*, 18(3) pp. 24-6.
- Patrick, J.D., Zhang, J. and Artola-Zubillaga, X. (1999). An Architecture and Query Language for a Federation of Heterogeneous Lexical and Dictionary Databases. *Computers and the Humanities* (in press).
- Patton, M.Q. (1990). *Qualitative Evaluation Methods*. Thousand Oaks: Sage Publications.
- Peräkylä, A. (1997). Reliability and Validity in Research Based on Tapes and Transcripts. In D. Silverman (Ed.), *Qualitative Research: Theory, Method and Practice*. London: Sage Publications.
- Petty L.C. and Rosen, E.F. (1987). Computer-based interactive video systems. *Behavior Research Methods, Instruments, & Computers*. 19(2), pp. 160-166.
- Pidgeon N. and Henwood, K. (1996). Grounded theory: practical implementation. In J.T.E. Richardson (Ed.), *Handbook of Qualitative Research Methods for Psychology and Social Sciences*. Leicester: The British Psychological Society.
- Potter, J. (1996). Discourse analysis and constructionist approaches: theoretical background. In J.T.E. Richardson (Ed.), *Handbook of Qualitative Research Methods for Psychology and Social Sciences*. Leicester: The British Psychological Society.
- Pressman, R.S. (1997). *Software Engineering: A Practitioner's Approach*. McGraw-Hill Companies.
- Psychology Software News (1995). *Newsletter of the CTI Centre for Psychology*, Volume 6, Number 1.
- QSR Software (1998). QSR NUDIST 4 Overview. <http://www.qsr.com.au/Software/N4/qsrn4.htm>, accessed 22/06/1998.
- Ragin, C.C. (1994). *Constructing Social Research*. Thousand Oaks: Pine Forge Press.
- Ray, W.J. (1993). *Methods: Towards a Science of Behaviour and Experience*. Pacific Grove, California: Brooks/Cole Publishing Company.

Reichardt, C.S. and Rallis, S.F. (1994). The Qualitative-Quantitative Debate: New Perspectives. In W.R. Shadish (Ed). San Francisco: Jossey-Bass Publishers.

Richards, L. (1999). *Using NVivo in Qualitative Research*. Melbourne: Qualitative Solutions and Research Pty.

Richards, L. and Richards, T. (1991). The Transformation of Qualitative Method: Computational Paradigms and Research Processes. In N.G. Fielding and R.M. Lee (Eds.), *Using Computers in Qualitative Research*. London: Sage Publications.

Richards, T.J. and Richards, L. (1994). Using Computers in Qualitative Research. In N.K. Denzin and Y.S. Lincoln (Eds.), *Handbook of Qualitative Research*. Thousand Oaks: Sage Publications.

Richards, T. and Richards, L. (1995). Using Hierarchical Categories in Qualitative Data Analysis. In U. Kelle (Ed.), *Computer-Aided Qualitative Data Analysis*. Sage Publications.

Roddick, J.F. and Patrick, J.D. (1992). Temporal Semantics in Information Systems – A Survey. *Information Systems*. 17 (3), pp. 249-267.

Roschelle, J. and Goldman, S. (1991). VideoNoter: A productivity tool for video data analysis. *Behavior Research Methods, Instruments, & Computers*. 23 (2), pp. 219-224.

Rose, H. (1991). Case Studies. In A. Graham and C. Skinner (Eds.), *Handbook for Research Students in the Social Sciences*. London: The Falmer Press,

Rosnow, R.L. and Rosenthal, R. (1996). *Beginning Behavioral Research, A Conceptual Primer*. Englewood Cliffs, New Jersey: Prentice Hall.

Ruhleder, K. and Jordan, B. (1998). *Capturing Complex, Distributed Activities: Video-Based Interaction Analysis as a Component of Workplace Ethnography*. <http://alexia.lis.uinc.edu/~ruhleder/publications/97.IFIPWG82.html>, accessed on 24/06/1998.

Ryan, C.C. and Russell, R.E. (1994). METACoder for Windows: real-time and multi-pass event logging and analysis in the social and behavioural sciences. *Psychology Software News*. Vol 5, Number 2.

Sanderson, P. (1998). *MacSHAPA: software for observational data analysis*. <http://www.aviation.uiuc.edu/institute/acdProg/epjp/MacShapa.html>, accessed 09/06/1998.

Sanderson, P.M. and Fisher, C. (1994). Exploratory Sequential Data Analysis: Foundations. *Human-Computer Interaction*. Vol.9, pp. 251-317.

Sanderson, P.M. and Fisher, C. (1997). Exploratory Sequential Data Analysis: Qualitative and Quantitative Handling of Continuous Observational Data. In G. Salvendy (Ed.), *Handbook of human factors and ergonomics*. New York: John Wiley & Sons.

- Sanderson, P., McNeese, M.D. and Zaff, B.S. (1994). Handling complex real-world data with two cognitive engineering tools: COGENT and MacSHAPA. *Behavior Research Methods, Instruments, & Computers*. 26(2), pp. 117-124.
- Sanderson, P.M., Scott, J., Johnston, T., Mainzer, J., Watanabe, L. and James, J. (1994). MacSHAPA and the enterprise of exploratory sequential data analysis (ESDA). *International Journal Human-Computer Studies*. 41, pp. 633-681.
- Schroeter, E. and Patrick, J.D. (1996). An Application Interface with Multiple Workspaces: The Context Switcher. *Conference Proceedings OzCHI'96*. IEEEComputer Society Press, Los Alamitos.
- Seidel, J. (1991). Method and Madness in the Application of Computer Technology to Qualitative Data Analysis. In N.G. Fielding and R.M. Lee (Eds.), *Using Computers in Qualitative Research*. London: Sage Publications.
- Silberschatz, A., Korth, H.F. and Sudarshan, S. (1997). *Database System Concepts*, McGraw-Hill.
- Silverman, D. (1993). *Interpreting Qualitative Data: Methods for Analysing Talk, Text and Interaction*. London: Sage Publications.
- Skou, C.V. (1998). *Kit: Qualitative interview and therapy analyser*. <http://www.psy.aau.dk/~dolphin/programs.htm#kit>. accessed 22/06/98.
- Smeaton, A.F. (1997). Information Retrieval: Still Butting Heads with Natural Language Processing? In M.T. Pazienza (Ed.), *Information Extraction: A Multidisciplinary Approach to an Emerging Information Technology*. Frascati, Italy.
- Smith, J.A. (1996). Evolving issues for qualitative psychology, In J.T.E. Richardson (Ed.), *Handbook of Qualitative Research Methods for Psychology and Social Sciences*. Leicester: The British Psychological Society.
- Sommer B. and Sommer R. (1991). *A Practical Guide to Behavioural Research, Tools and Techniques*. New York: Oxford University Press.
- Stampe, D.M. (1993). Heuristic filtering and reliable calibration methods for video-based pupil-tracking systems. *Behavior Research Methods, Instruments, & Computers*. 25(2), 137-142.
- Stangor C. (1998). *Research Methods for the Behavioral Sciences*. Boston: Houghton Mifflin Company.
- Tapp, J. and Walden, T. (1993). PROCODER: A professional tape control, coding, and analysis system for behavioral research using video tape. *Behavior Research Methods, Instruments, & Computers*. 25(1), pp. 53-56.
- Tesch, R. (1990). *Qualitative Research: Analysis Types and Software Tools*. New York: The Falmer Press.

- Tesch, R. (1991). Software for Qualitative Researchers: Analysis Needs and Program Capabilities. In N.G. Fielding and R.M. Lee (Eds.), *Using Computers in Qualitative Research*. London: Sage Publications.
- The Dominion (1998). NZ Infotech Weekly. Issue No 348, 8 June 1998.
- Tindall, C. (1994). Issues of Evaluation. In P. Banister, E. Burman, I. Parker, M. Taylor, and C. Tindall (Eds.), *Qualitative Methods in Psychology: A Research Guide*. Buckingham: Open University Press.
- Tuckman, B.W. (1999). *Conducting Educational Research*. Fort Worth: Harcourt Brace College Publishers.
- VANNA (1998a). <http://www.dgp.toronto.edu/OTP/Techdocs/Vanna.html>, accessed 17/06/1998.
- Vlugt, van der, M.J., Kruk, M.R., Erp, van, A.M.M., Geuze, R.H. (1992). CAMERA: A system for fast and reliable acquisition of multiple ethological records. *Behavior Research Methods, Instruments, & Computers*. 24(2), pp. 147-149.
- Wahl, T. and Rothermel, K. (1994). Representing Time in Multimedia Systems. IEEE, ICMCS.
- Walsh, B., Lavalli, T. (1996). Beyond BeanCounting, Qualitative Research Software for Business. *MicroTimes*. <http://www.microtimes.com>, accessed 26/05/1999.
- Wardhaugh, R., (1995). *Understanding English Grammar: A linguistic Approach*. Cambridge, USA: Blackwell Publishers.
- Watson, C. et al. (1991). *Television Violence: an analysis of the portrayal of violent acts on the Three New Zealand broadcast television channels during the week 11th-17th February 1991*. Research project for the Broadcasting Standards Authority by the Educational Research and Development Centre, Massey University, New Zealand.
- Watts,L. (1998). Action Recorder - Behavioural Analysis Made Easy. <http://www.york.ac.uk/~law4>, accessed 17/07/1998.
- Watts, L. and Monk, A.F. (1996). Remote assistance: a view of the work and a view of the face? CHI'96, *Proceedings Conference on Human Factors in Computing Systems*. Vancouver, British Columbia, Canada.
- WCER (1998). <http://www.wcer.wisc.edu/multimedia/videoanalysis/videoanalysis.htm>, accessed 26/05/1999.
- Weaver, A. and Atkinson, P. (1994). *Microcomputing and Qualitative Data Analysis*. Aldershot: Avebury.
- Weber, K. and Poon, A. (1994). Marquee: A Tool for Real-Time Video Logging. *Proceedings CHI'94 Human Factors in Computing Systems*. Boston.

- Weitzman, E.B. and Miles, M.B. (1995). *Computer Programs for Qualitative Data Analysis: A Software Sourcebook*. Thousand Oaks: Sage Publishers.
- Whittington, C. J. (1999). *Cognitive deficits in Parkinson's disease*. Manuscript in preparation.
- Wilkinson, I. A. G. and Townsend, M. A. R. (1999). From Rata to Rimu: Grouping for instruction in 'best practice' New Zealand classrooms. *The Reading Teacher*. (submitted).
- Witter, D.I. and Berry, M.W. (1998). Downdating the Latent Semantic Indexing Model for Conceptual Information Retrieval. *Computer Science Journal*. Volume 41, Issue 8.
- Wolcott, H.F. (1994). *Transforming Qualitative Data: Description, Analysis, and Interpretation*. Thousand Oaks: Sage Publications.
- Yin, R. (1993). *Applications of Case Study Research*. Newbury Park, California: SAGE Publications.
- Yin, R.K. (1994). *Case Study Research: Design and Methods*. Applied Social Research Methods Series. Volume 5, Thousand Oaks, California: Sage Publications.

Appendix A

Summary of FSQL Query Options

Query Feature	Query Type	Construction	Evaluation based on	Set of results
subject - verb - object and concept - object relationships	basicQuery	words of the vocabulary combined using the grammar rules of the coding language	single code instances coding sentences semantic tree of basicQuery is subtree of semantic tree of code instance	result sentence constructed by single coding sentence one time/position value in one study file
hierarchies	basicQuery	parameter setting	hierarchical definition of words in vocabulary	result sentence constructed by single coding sentence one time/position value in one study file
word links	basicQuery	parameter setting	word link definition in vocabulary	result sentence constructed by single coding sentence one time/position value in one study file
concept relationships	basicQuery	parameter setting	relationships defined in concept building module	result sentence constructed by single coding sentence one time/position value in one study file
wild cards	basicQuery	use of wild cards in place of words ANY-PERSON/THING ANY-ACTIVITY ANY-CONCEPT ANY-CONJUNCTION ALL-DESCS	vocabulary definition	result sentence constructed by single coding sentence one time/position value in one study file
boolean NOT	booleanQuery	NOT basicQuery	set of code instances coding sentences code instances fulfilling basicQuery are not considered for result	result sentence constructed by single coding sentence one time/position value in one study file
boolean AND	booleanQuery	subQuery1 AND subQuery2	set of code instances coding sentences code instances fulfilling both subQuery1 and subQuery2	result sentence constructed by single coding sentence one time/position value in one study file
boolean OR	booleanQuery	subQuery1 OR subQuery2	set of code instances coding sentences code instances fulfilling either subQuery1 or subQuery2 or both	result sentence constructed by single coding sentence one time/position value in one study file
time/position based period	fullQuery	subQuery FROM tp1 TO tp2	set of code instances time/position value code instances fulfilling subQuery and have time/position values between tp1 and tp2	result sentence constructed by single coding sentence one time/position value in one study file

time/position based comparison 'before'	fullQuery	subQuery1 BEFORE subQuery2 WITHIN tp	set of code instances time/position value all pairs (ci1, ci2) of code instances fulfilling subQuery1 and subQuery2 where the start time/position value of ci1 is less than the start time/position value of ci2 and the two time/position values are within a period of tp	result sentence constructed by two coding sentences: coding sentence1 BEFORE coding sentence2 one time/position value in one studyFile: period which covers both time/position values
time/position based comparison 'same-as'	fullQuery	subQuery1 SAME-AS subQuery2 WITHIN tp	set of code instances time/position value all pairs (ci1, ci2) of code instances fulfilling subQuery1 and subQuery2 where the two time/position periods of ci1 and ci2 overlap or are a maximum distance of tp apart	result sentence constructed by two coding sentences: coding sentence1 SAME-AS coding sentence2 one time/position value in one study file: period which covers both time/position values
time/position based comparison 'after'	fullQuery	subQuery1 AFTER subQuery2 WITHIN tp	set of code instances time/position value all pairs (ci1, ci2) of code instances fulfilling subQuery1 and subQuery2 where the end time/position value of ci1 is greater than the end time/position value of ci2 and the two time/position values are within a period of tp	result sentence constructed by two coding sentences: coding sentence1 AFTER coding sentence2 one time/position value in one study file: period which covers both time/position values
sequence based	fullQuery	subQuery1 FOLLOWED-BY subQuery2	set of code instances time/position value all pairs (ci1, ci2) of code instances fulfilling subQuery1 and subQuery2 constructed by considering the ci1s in sequence of increasing time/position values and pairing them with the next ci2 with a greater time/position value	result sentence constructed by two coding sentences: coding sentence1 FOLLOWED-BY coding sentence2 one time/position value in one study file: period which covers both time/position values

combination across studyFiles	fullQuery	subQuery1 COMBINED-WITH subQuery2 VIA segmentCode	set of code instances time/position value segmentCode is a word of the category concept which is the root of a hierarchy of concept words; the words of this hierarchy have been used to code related behaviour captured in different studyFiles all pairs (ci1, ci2) of code instances fulfilling subQuery1 and subQuery2 where for each concept word cw belonging to the segmentCode hierarchy the time/position value of ci1 is within the time/position value of cw for the studyFile ci1 relates to and the time/position value of ci2 is within the time/position value of cw for the studyFile ci2 relates to	result sentence constructed by two coding sentences: coding sentence1 COMBINED-WITH coding sentence2 VIA concept word two time/position values relating to two study files
--------------------------------------	-----------	---	--	--

The term 'subQuery' states where a query feature can be applied to either a basic query or to a Boolean query. 'tp1' and 'tp2' indicate time or position values.

Appendix B

Examples for FSQL Query Matching Algorithm

Set of Code Instances1

Coding sentence: Jim hit ball with bat and fast run to Bill

Semantic tree pattern: 1233223

Full information:

Tree Level	Words	Categories
1	Jim	P
2	hit	A
3	ball	P
3	with bat	RP
2	and	K
2	fast run	DA
3	to Bill	RP

Query1

Query sentence: Jim run

Basic query tree pattern: 12

Full information:

Tree Level	Words	Categories
1	Jim	P
2	run	A

Steps in the matching algorithm

	Query Tree Pattern	Semantic Tree Pattern
1 . Identify first match in tree patterns:	12 *	1233223 *
2. Test word match:	Jim	Jim
3. Word match, advance position in query tree pattern	12 *	1233223 *
4 . Identify next match in semantic tree pattern	12 *	1233223 *
5. Test word match:	run	hit
6. No word match; find next match in semantic tree pattern	12 *	1233223 *
7. Test word match:	run	and
8. No word match; find next match in semantic tree pattern	12 *	1233223 *
9. Test word match:	run	run
10. Word match, end of query tree pattern, match between query and code instances identified, formulate query instance		

Query2

Query sentence: Junior Student run

Basic query tree pattern: 12

Full information:

Tree Level	Words	Categories
1	Junior Student	P
2	run	A

Steps in the matching algorithm

	Query Tree Pattern	Semantic Tree Pattern
1 . Identify first match in tree patterns:	12 *	1233223 *
2. Test word match:	Junior Student	Jim
3. Match based on word hierarchies, <i>continued as in query1 ...</i>		

Query3

Query sentence: ANY-PERSON/THING run

Basic query tree pattern: 12

Full information:

Tree Level	Words	Categories
1	ANY-PERSON/THING	P
2	run	A

Steps in the matching algorithm

	Query Tree Pattern	Semantic Tree Pattern
1 . Identify first match in tree patterns:	12 *	1233223 *
2. Test word match:	ANY- PERSON/ THING	Jim
3. Match on category basis, <i>continued as in query1 ...</i>		

Set of Code Instance2

Coding sentence: Jim hit ball with bat and Bill fast run

Semantic tree pattern: 1233112

Full information:

Tree Level	Words	Categories
1	Jim	P
2	hit	A
3	ball	P
3	with bat	RP
1	and	K
1	Bill	P
2	fast run	DA

Query

Query sentence: Jim run

Basic query tree pattern: 12

Full information:

Tree Level	Words	Categories
1	Jim	P
2	run	A

Steps in the matching algorithm

	Query Tree Pattern	Semantic Tree Pattern
1 . Identify first match in tree patterns:	12 *	1233112 *
2. Word match, advance position in query tree pattern	12 *	1233223 *
3 . Identify next match in semantic tree pattern	12 *	1233223 *
4. Test word match:	run	hit
5. No word match; find next match in semantic tree pattern without passing any higher level position	12 *	1233112
6. No match in semantic tree pattern, backtracking to previous position in basic query tree pattern and identification of match	12 *	1233112 *
7. Test word match:	Jim	and
8. No word match; find next match in semantic tree pattern	12 *	1233112 *
9. Test word match:	Jim	Bill
10. No word match; no next match in tree patterns; no match between query and code instances		

Appendix C

The Windows Belonging to Each Module

The following table indicates how the modules of the application PAC have been structured. The elements of each module, the main window and the additional windows, are named and their tasks are briefly explained.

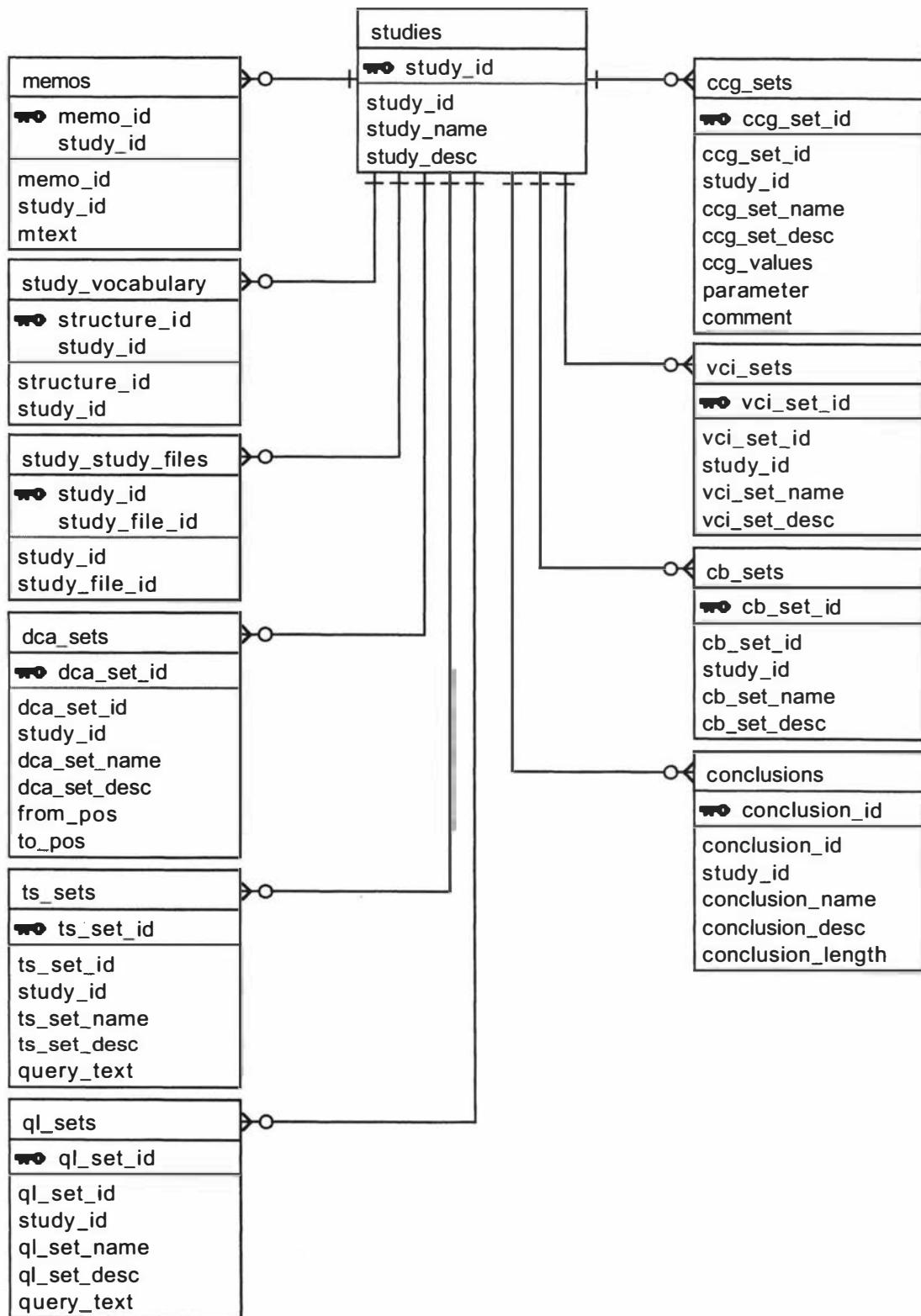
Window	Task
Module StudyFiles	
Main Window	Identification of behaviour recordings in the file system Viewing behaviour recordings Specification of name and description for use of behaviour recordings in PAC
Window FileViewer (multiple windows)	Display of video or text files
Module Vocabulary	
Main Window	Definition of vocabulary on domain basis Definition of hierarchies within vocabulary
Window WordLinks	Definition of links between words of the vocabulary
Module Studies	
Main Window	Definition of study
Window Select/Remove StudyFiles	Definition of which study files are used in specific study
Window Select/Remove Vocabulary	Definition of which words of the vocabulary are used in specific study
Module Description	
Main Window	Definition of description set by giving name and description and selecting study files to be used
Window VocabularyAnnotation	Selection of subset of words from study vocabulary used for current coding Definition of macros Definition of coding prefix Formulating of coding sentence Initiating of annotation input
Window QuerySetsStudyFiles	Selection which of the study files defined for description set is used for current coding or annotation Setting of parameters for description process relating to presentation of study files and formulation of coding sentences Not implemented: Selection of existing query set for presentation of only selected study file sequences
Window DescriptionsCodes	Display of existing code instances for description set Display of newly formulated code instances Modification of code instances Submit of code instances to database
Window Annotations	Display of existing annotations for description set Display of newly formulated annotations Modification of annotations Submit of annotations to database
Module QueryLanguage	
Main Window	Definition of query set by giving name and description and selecting description sets to be used to apply queries to
Window Formulate Query	Formulate the query to be applied to the code instances stored in the selected description sets Setting of parameters for query evaluation
Window QueryResults	Display of query results Submit of query results to database
Module TextSearch	
Main Window	Definition of text search set by giving name and description and selecting description sets to be used to apply text searches to
Window FormulateSearch	Formulate the text search to be applied to the

	annotations stored in the selected description sets Setting of parameters for text search evaluation
Window Search Results	Display of text search results Submit of text search results to database
Module Counts/Calculation/Graphs	
Main Window	Definition of counts, calculations, graphs set by giving name and description and selecting query and text search sets to be used to apply interpretation to
Window Calculations	Display of counts, duration, frequency values for query and text search sets selected for set Selection of individual query or text search sets for comparison of values in graph or table format Setting of parameters for comparisons
Window CalculationResults	Display of comparison results Submit of comparison results to database
Module Viewing/Comparison/ImageAnalysis	
Main Window	Definition of viewing, comparison, image analysis sets by giving name and description and selecting query and text search sets to be used to apply interpretation to
Window ViewingComparison	Display of description text and position values of individual query and text search results stored in query and text search sets selected for set Selection of individual query or text search results for display of associated segment of behaviour recording Submit of sequence of selected query or text search results to database Not sufficiently implemented: Use of image analysis tools for video recordings
Window FileViewer (multiple windows)	Display of video or text file
Window ImageAnalysis	Selection of graphical annotation tools
Module ConceptBuilding	
Main Window	Definition of concept building set by giving name and description
Window ConceptGraph	Display of words of the concept category of the study vocabulary Definition of concept relationships Submit of relationships to database
Module Conclusion	
Main Window	Definition of conclusion set by giving name and description and selecting counts, calculation, graphs sets, viewing, image analysis sets and concept building sets to be used to support conclusion statement
Window ConclusionConstruction	Definition of segments of conclusion
Window ConclusionSegment (multiple windows)	Entering of text of conclusion statement for segment Selection of supporting data
Window ConclusionReplay	Replay of conclusion segments displaying conclusion statement and supporting data

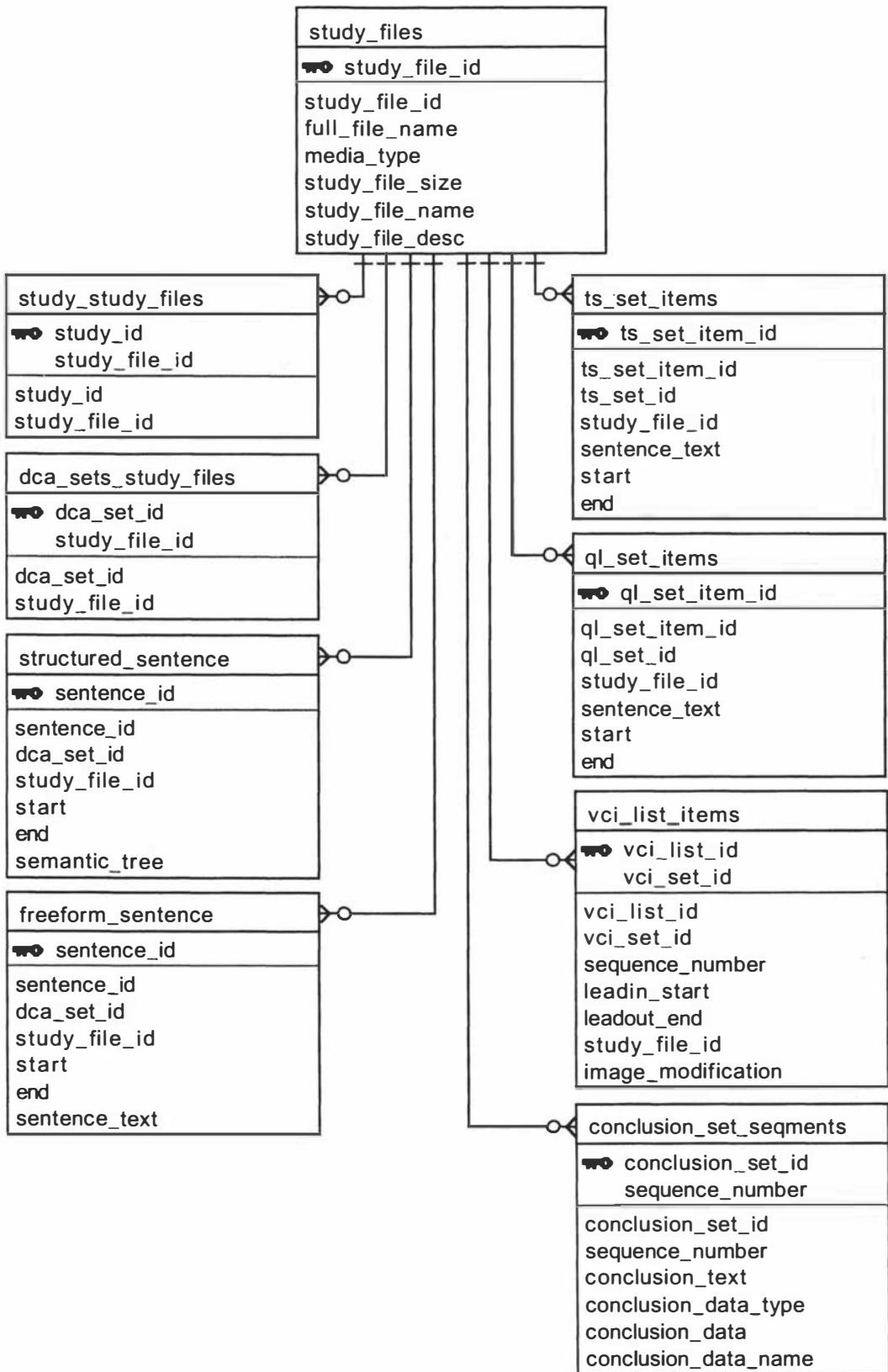
Appendix D

The PAC Database Tables

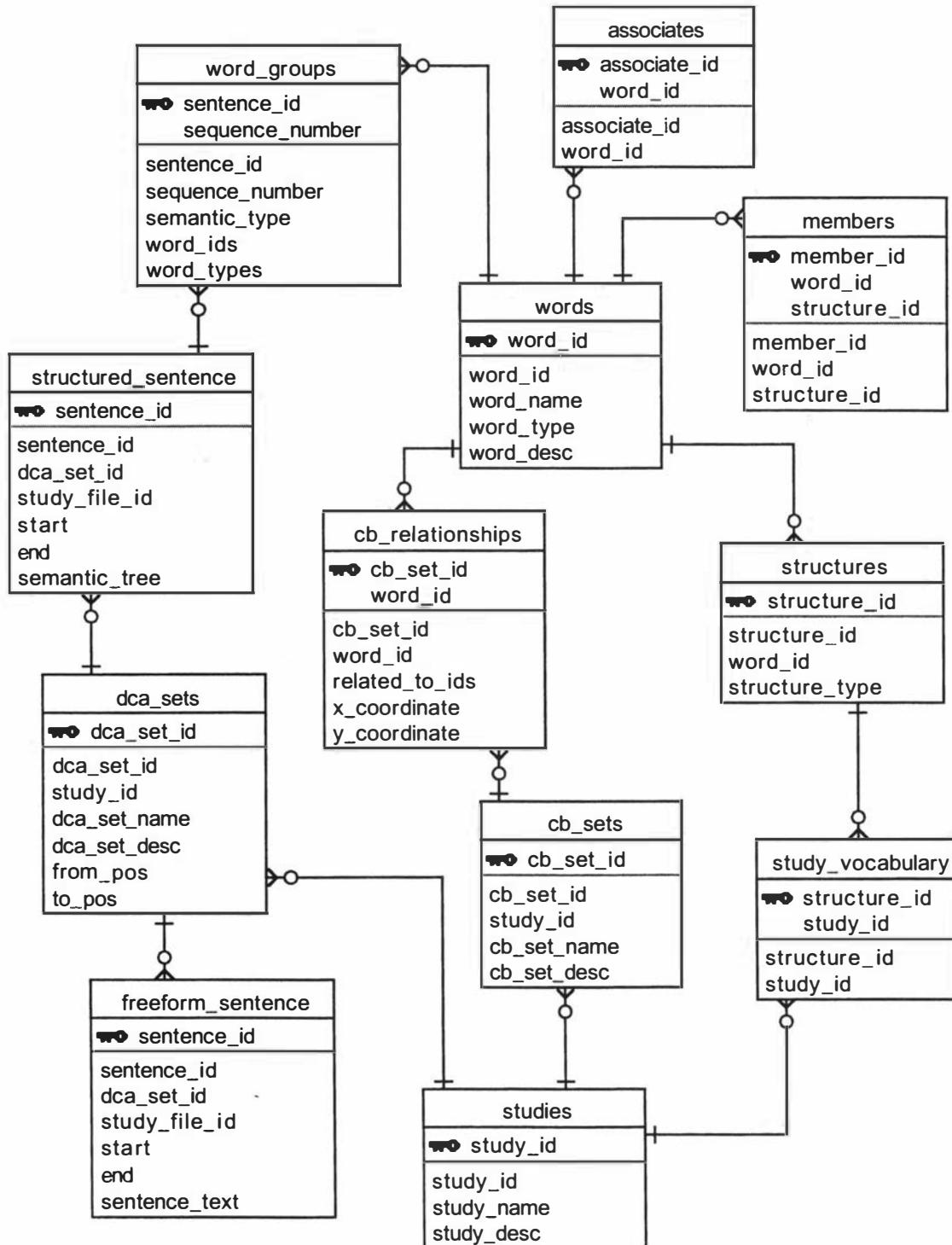
PAC Database Tables related to Study Module



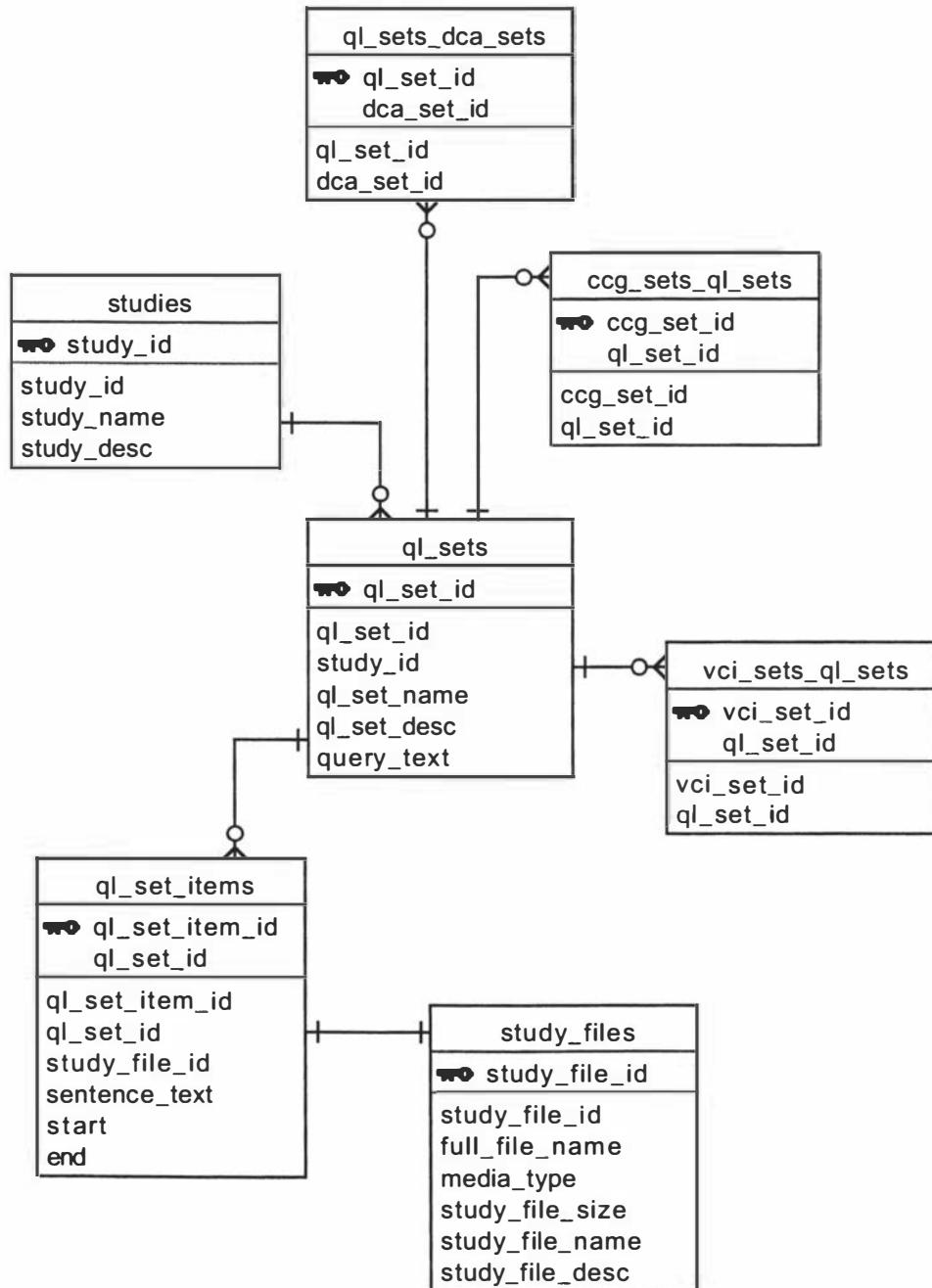
PAC Database Tables related to StudyFiles Module



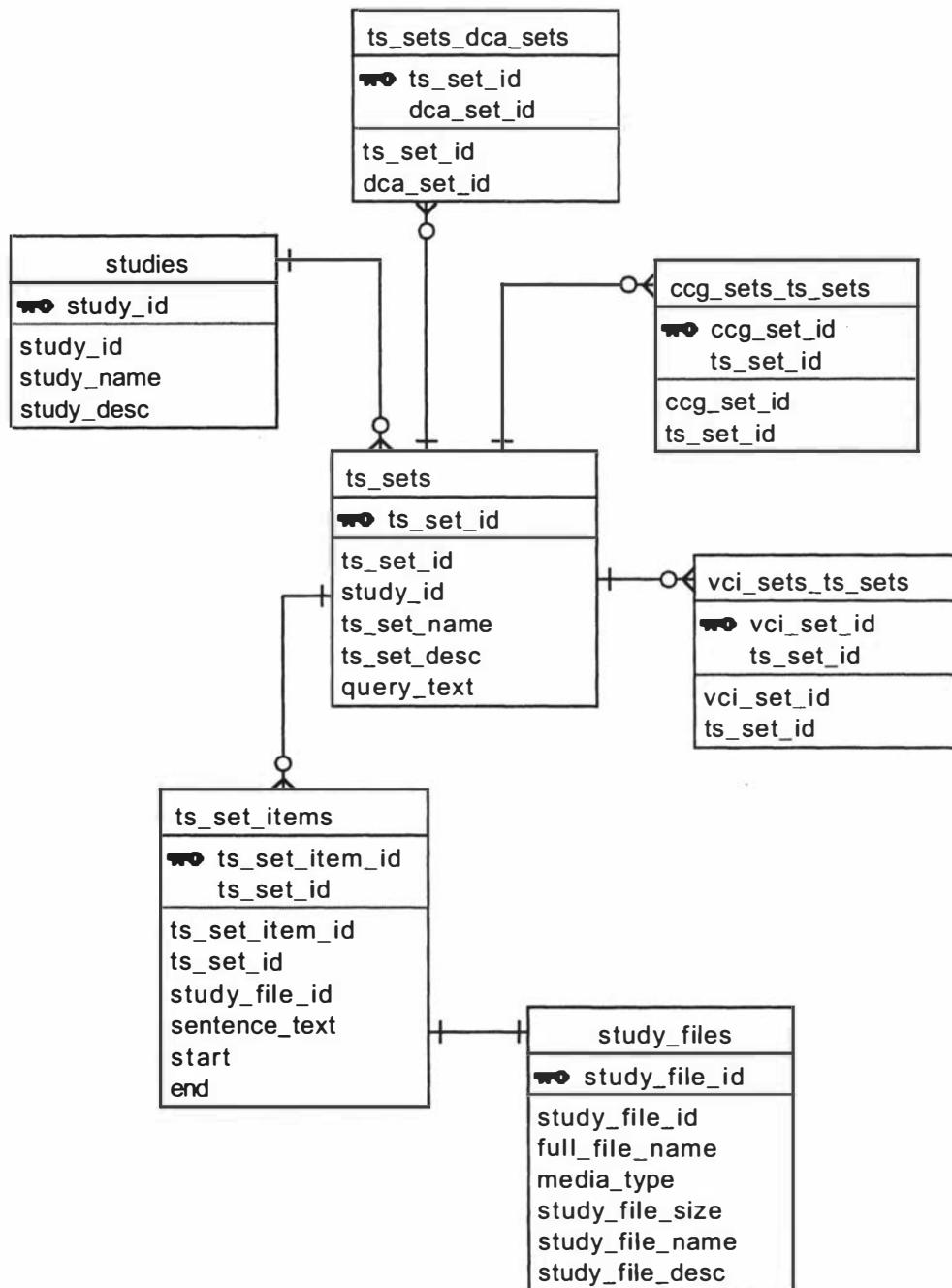
PAC Database Tables related to Description and ConceptBuilding Modules



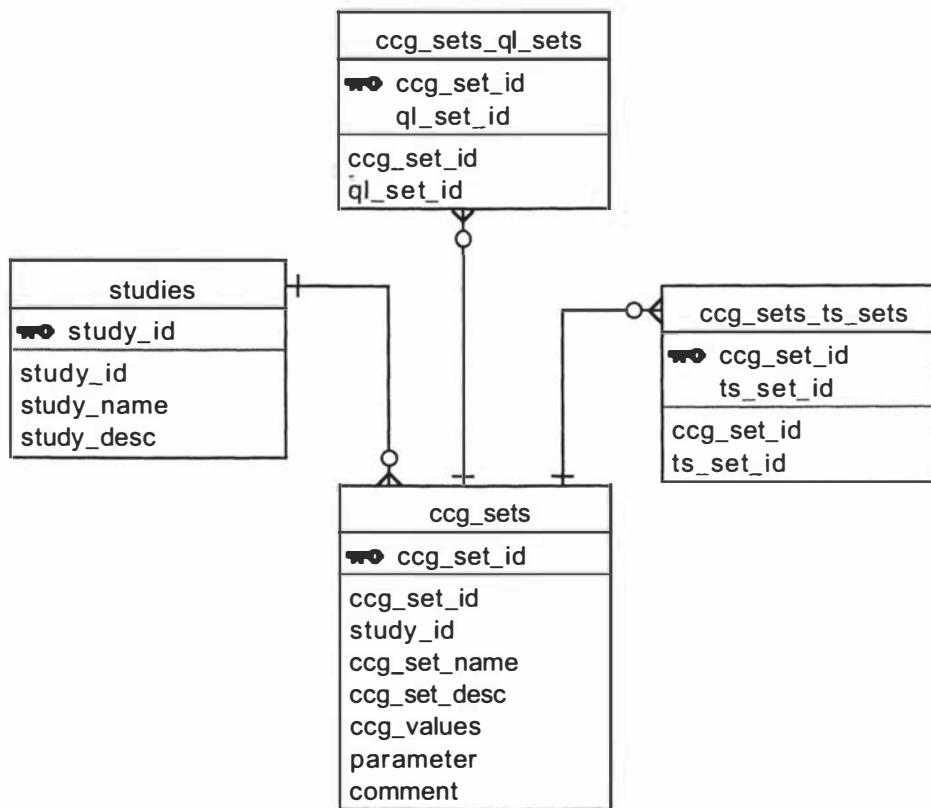
PAC Database Tables related to QueryLanguage Module



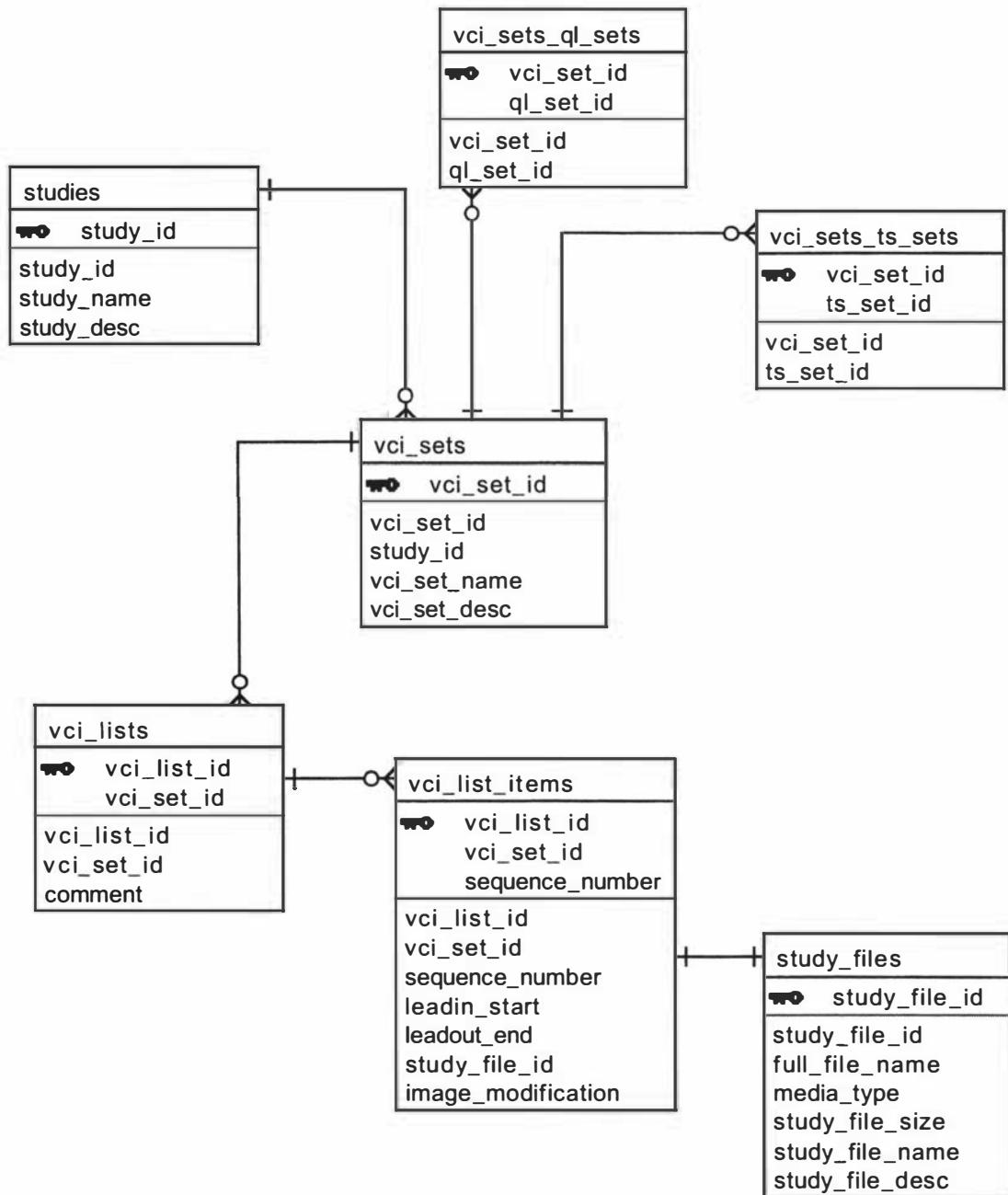
PAC Database Tables related to TextSearch Module



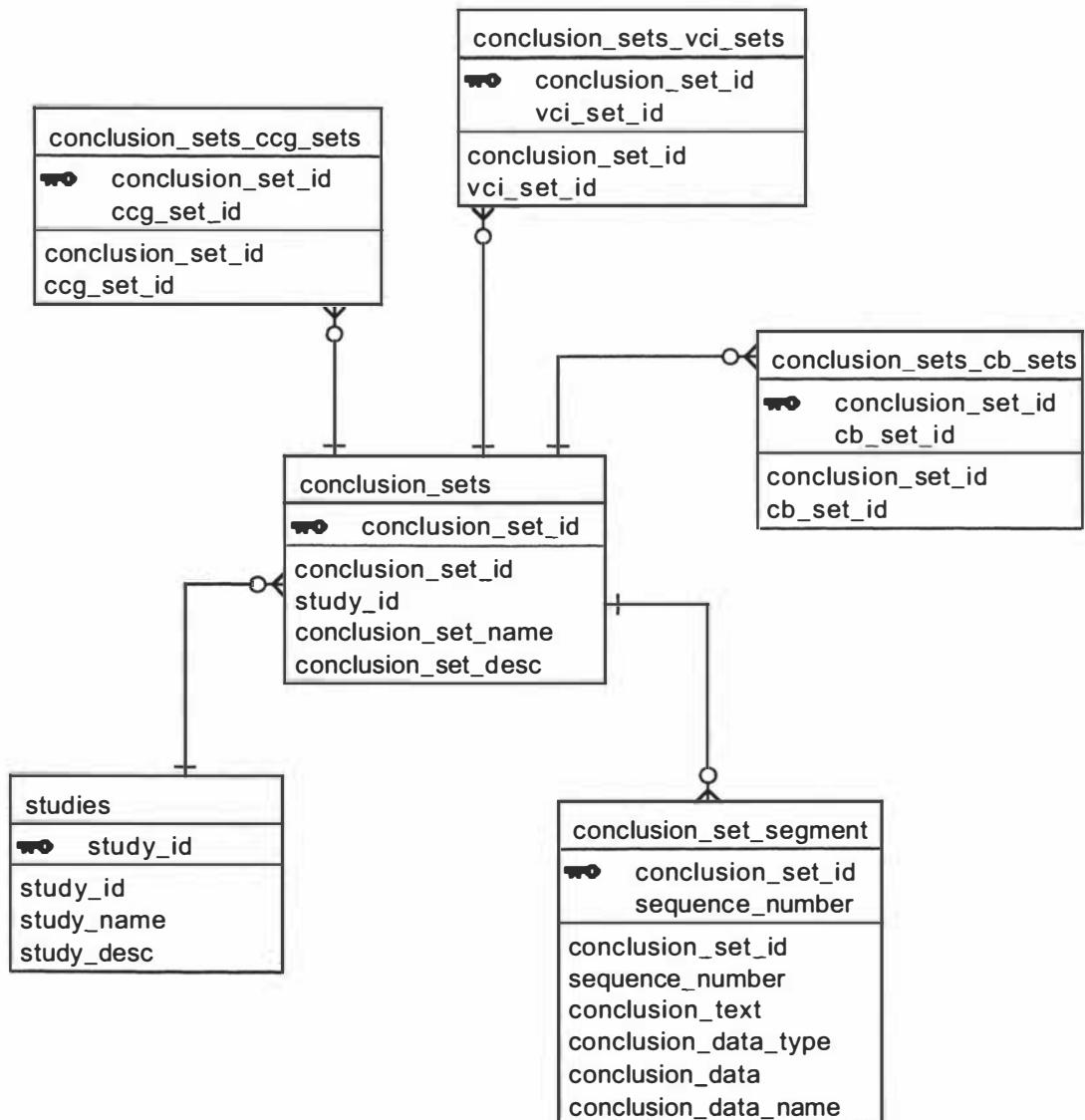
PAC Database Tables related to Count/Calculation/Graphs Module



PAC Database Tables related to Viewing/Comparison/ImageAnalysis Module



PAC Database Tables related to Conclusion Module



PAC Database Tables Application Wide

reference_files
reference_file_id
reference_file_id
full_file_name
media_type
reference_file_size
reference_file_name
reference_file_desc

notes
which
which
notes_text

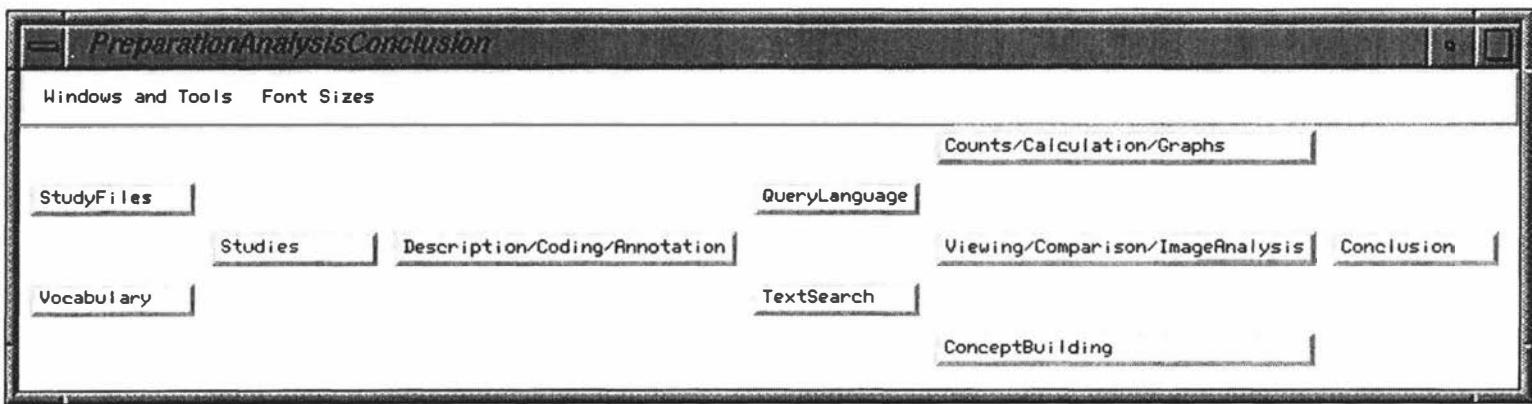
Appendix E

Screenshots from the Implementation PAC

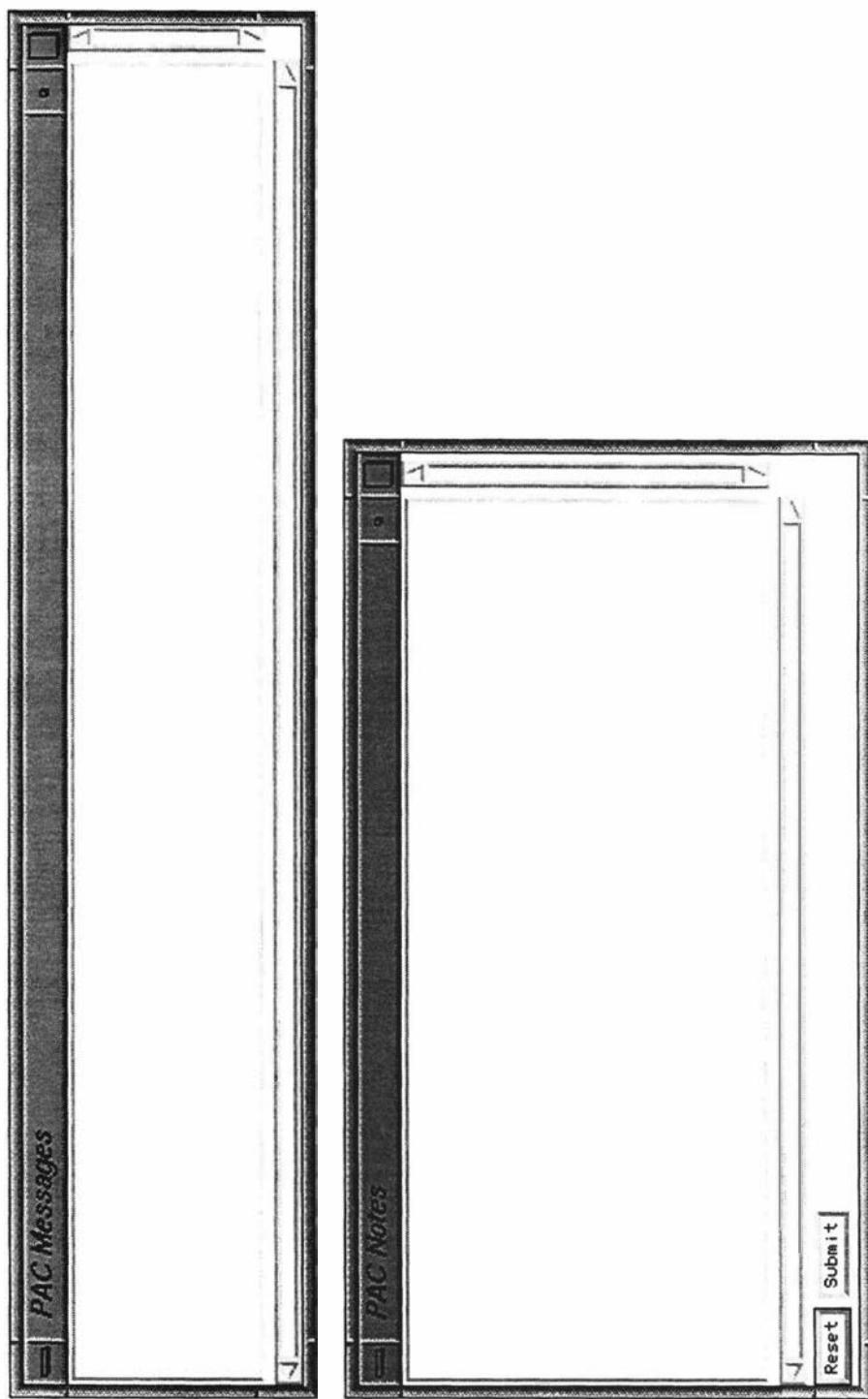
This appendix presents screenshots from the application PAC. For each module screenshots for the main window and for the supplementary windows are shown. The data displayed in the screenshots are from the three case studies Reading Groups, Dog Calming and Parkinson's disease.

For the modules StudyFiles, Vocabulary, Studies, Description, QueryLanguage and TextSearch the data presented are directly taken from the case studies. For the remaining modules, Counts/Calculations/Graphs, Viewing/Comparison/ImageAnalysis, ConceptBuilding and Conclusion the data presented are based on the case study data but have been produced after conclusion of the case studies for the purpose of presenting features of these modules.

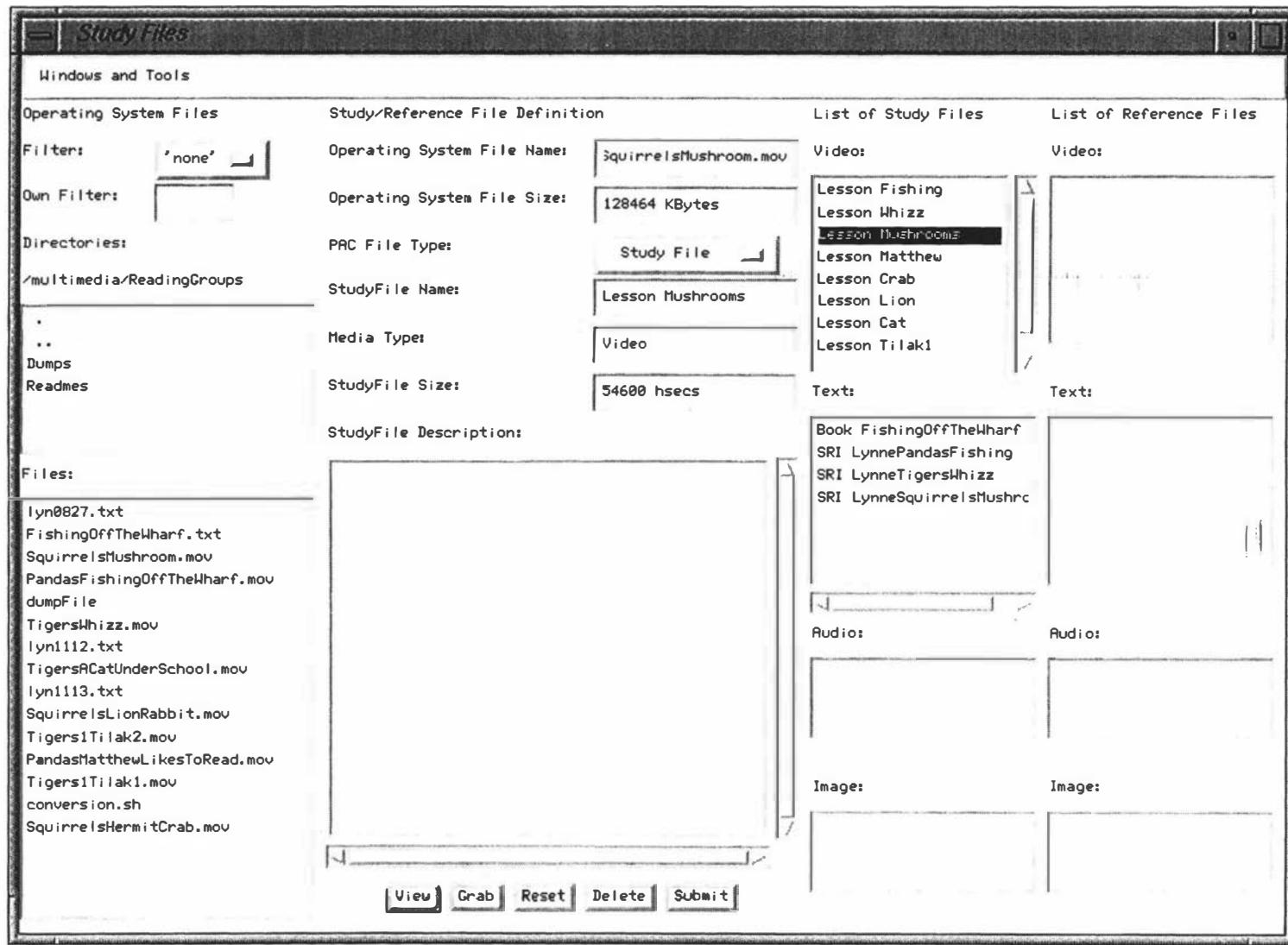
The Application PAC: Main Window



The Application PAC: Windows Messages and Notes



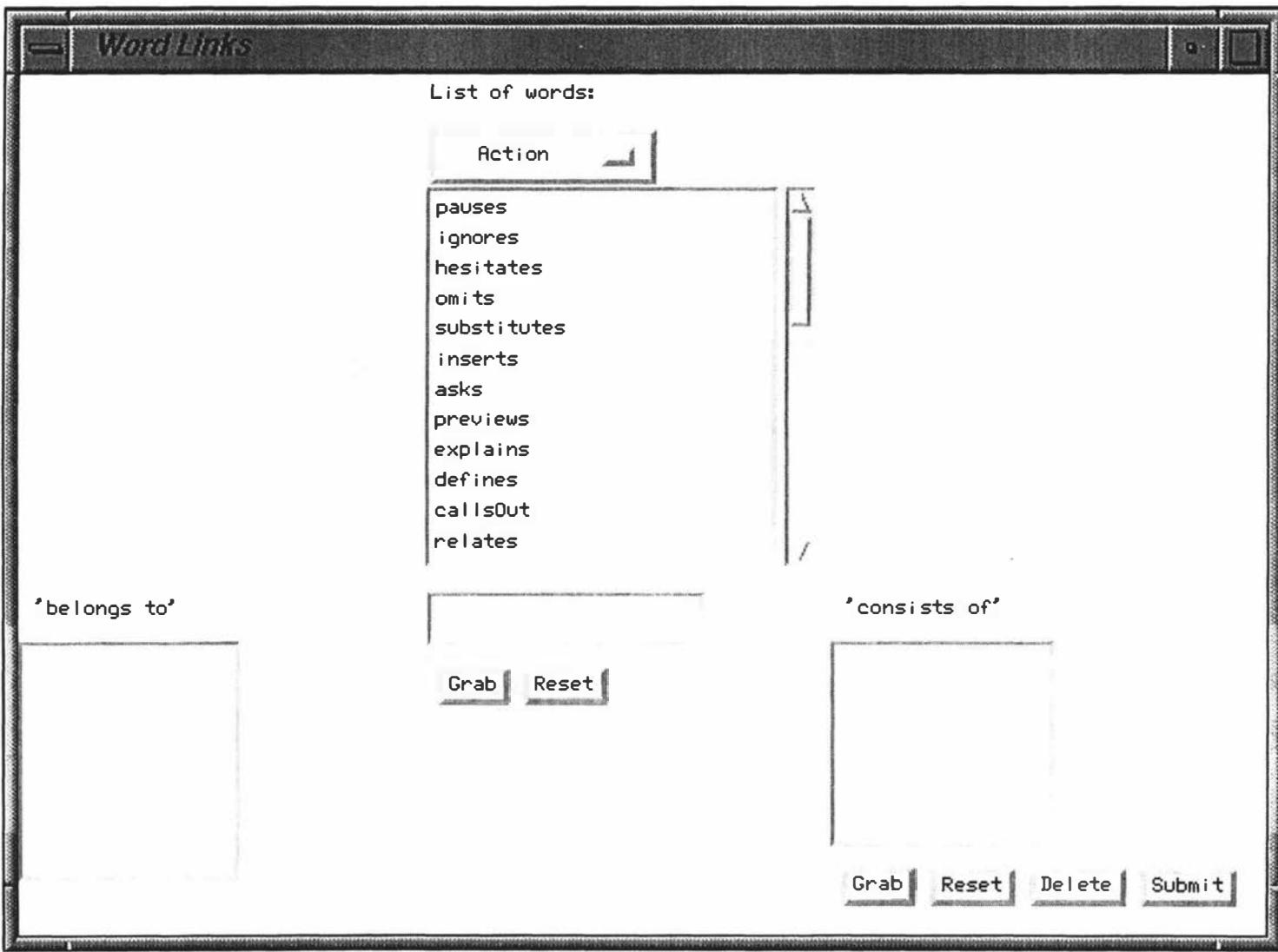
Module StudyFiles: Main Window (Reading Groups Case Study)

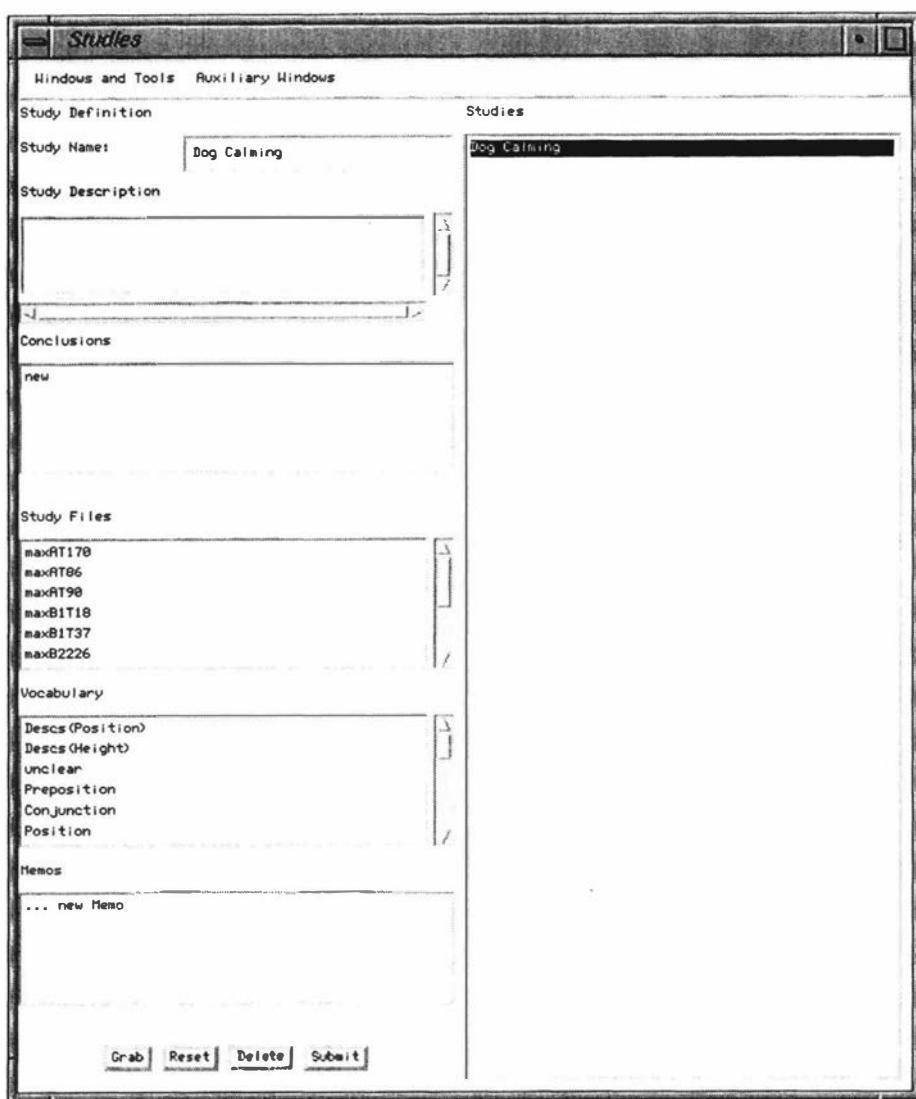


Module Vocabulary: Main Window (Reading Groups Case Study)

Windows and Tools		List of words:		Vocabulary:
Word Definition:		Action:		Person/Thing: Reader OtherChildren Group Children Teachers AnotherChild
Word name:	Person/Thing	Places:	ignores has its on its substitutes inserts assists previous explains defines causes relates directs/intention reads	Concepts: Words in Lion Words in Fishing Attention Codes Substantive engagement scale Sustained engagement scale
Word type:		Activities:		
Word definitions:		Auxiliaries:		
		recorders:		
		discusses:		
		makes:		
		reminds:		
		self/corrects:		
		says:		
		gives:		
		tells:		
		paraphrases:		
		resolves:		
		calls for help:		
		shows:		
		confirms:		
		disconfirms:		
		points to:		
		Conjunctions:		
		ConJus:		
		Prepositions:		
		Preps:		
		Description:		ReadingDesc PositionDesc CloseDesc ErrorDesc TimingDesc AttributeDesc
		Reset Structure Elements:	Reset Delete Send	
		Grab:		

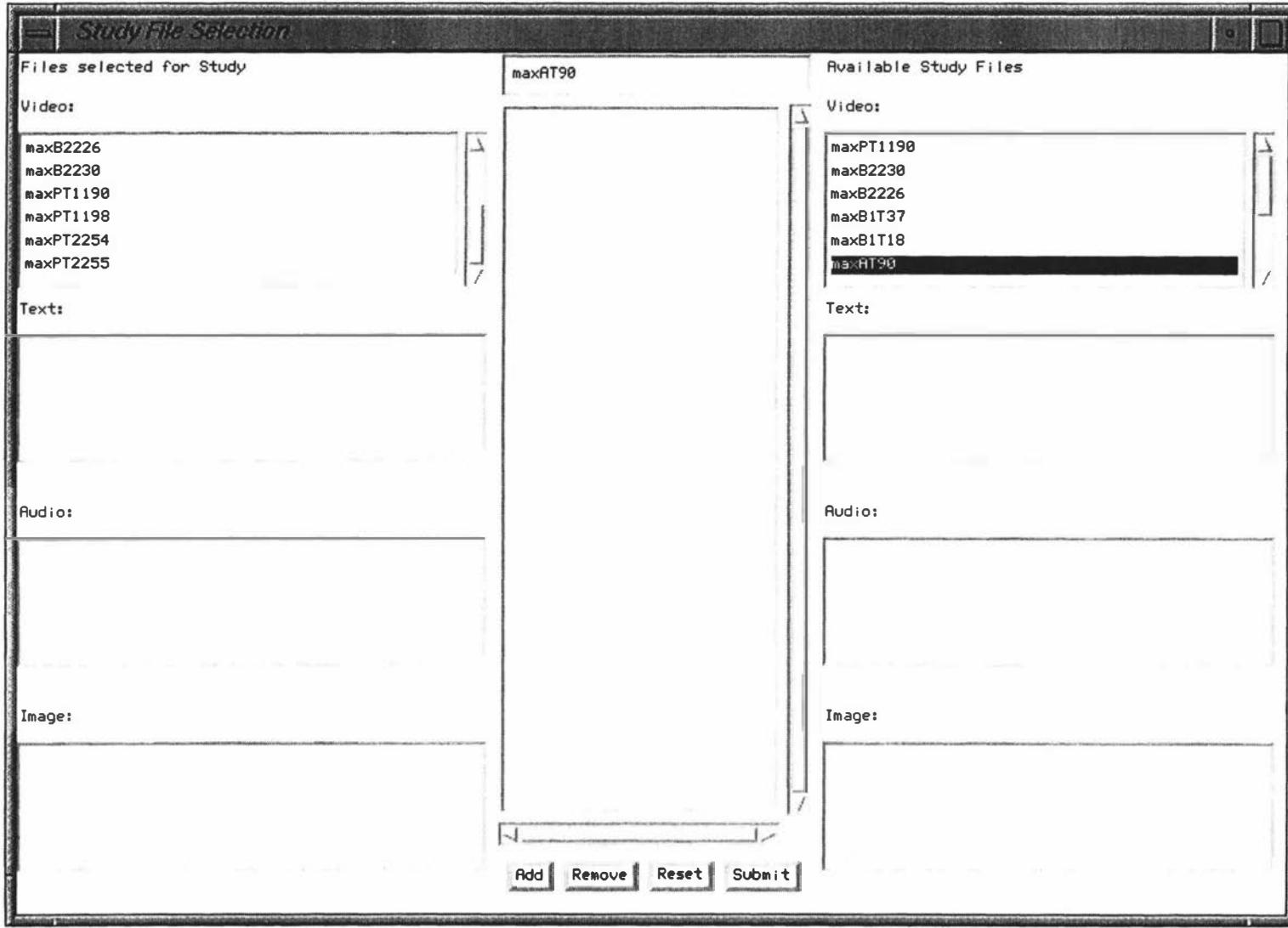
Module Vocabulary: Window Word Links (Reading Groups Case Study)

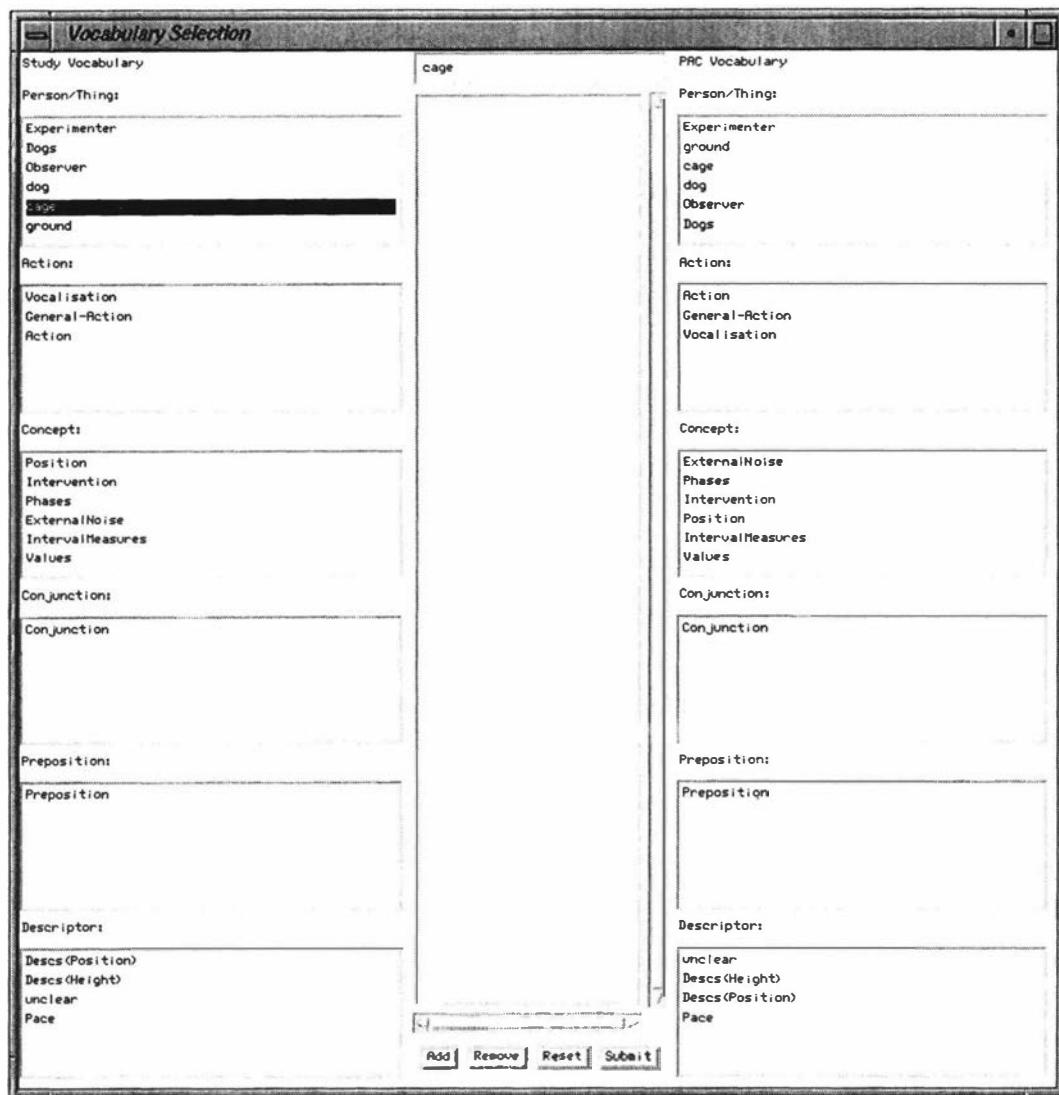


Module Studies: Main Window (Dog Calming Case Study)

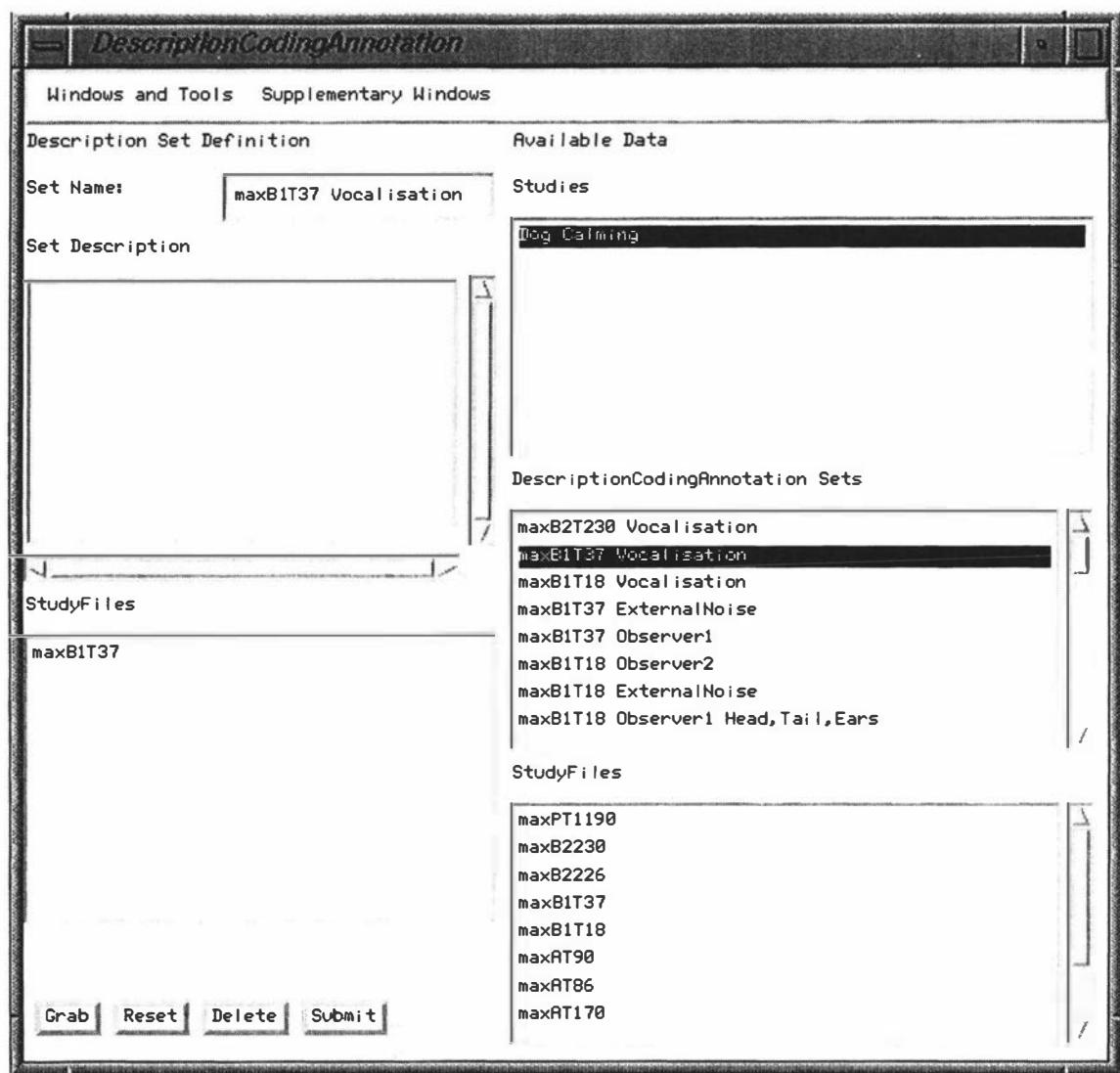
APPENDIX E

Module Studies: Window Study File Selection (Dog Calming Case Study)

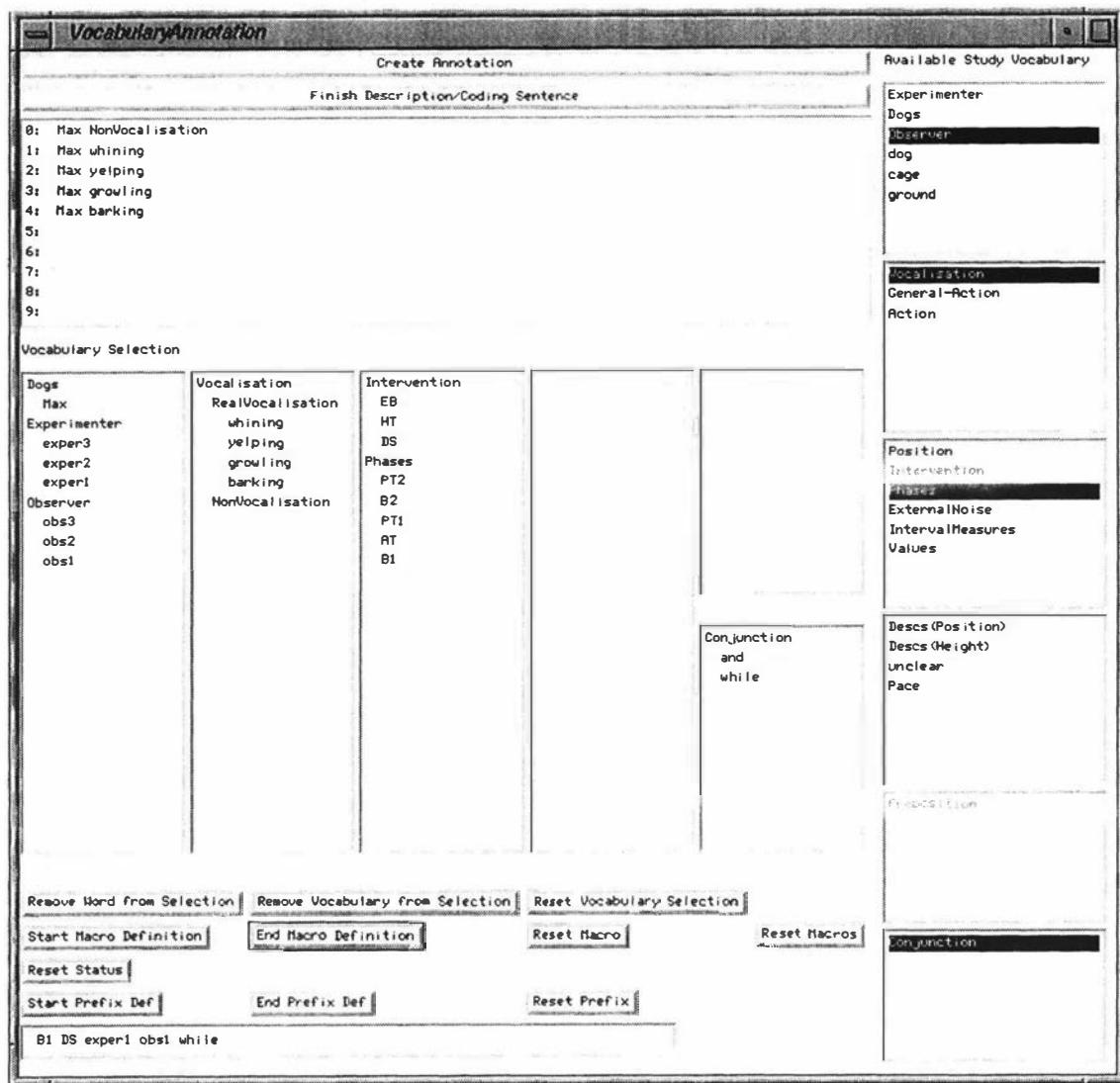


Module Studies: Window Vocabulary Selection (Dog Calming Case Study)

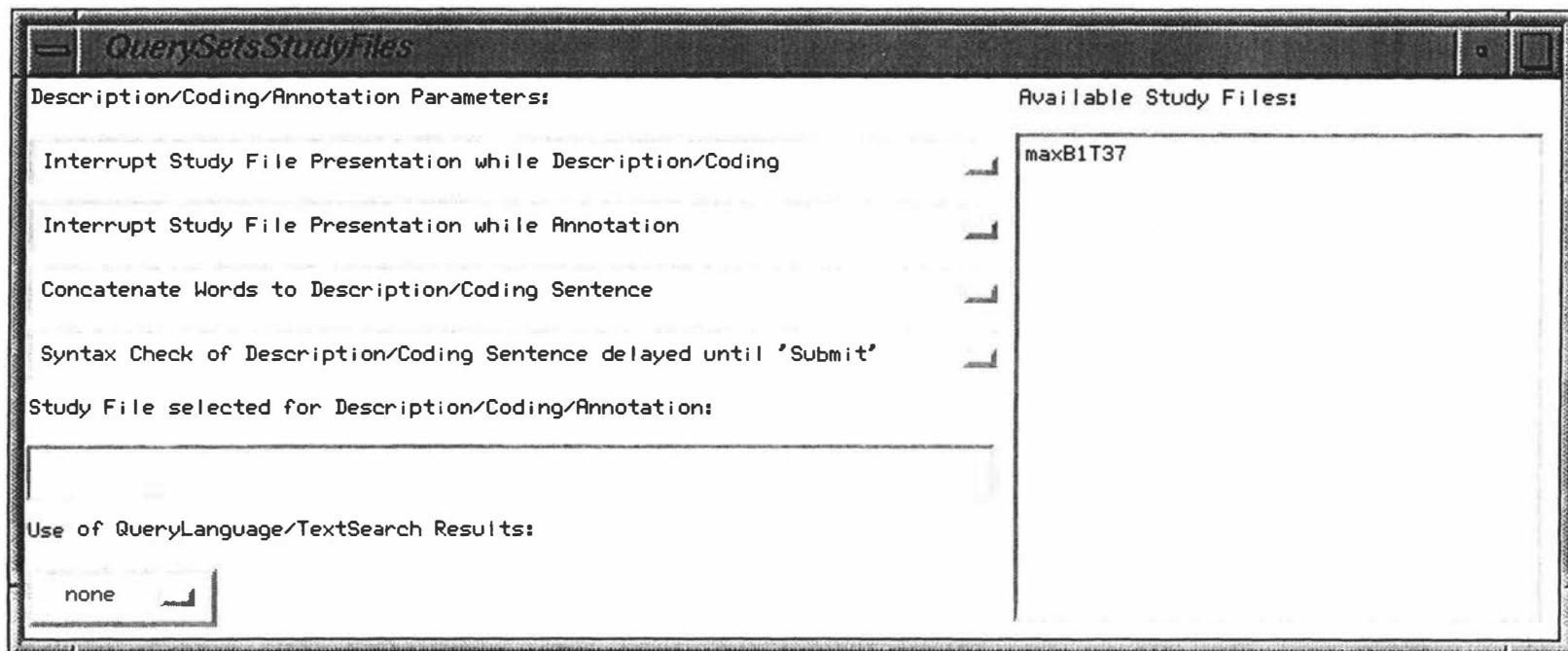
Module Description: Main Window (Dog Calming Case Study)



Module Description: Window Vocabulary Annotation (Dog Calming Case Study)



Module Description: Window Query Sets Study Files (Dog Calming Case Study)



Module Description: Window Descriptions Codes (Dog Calming Case Study)

Descriptions/Codes

Existing Descriptions/Codes

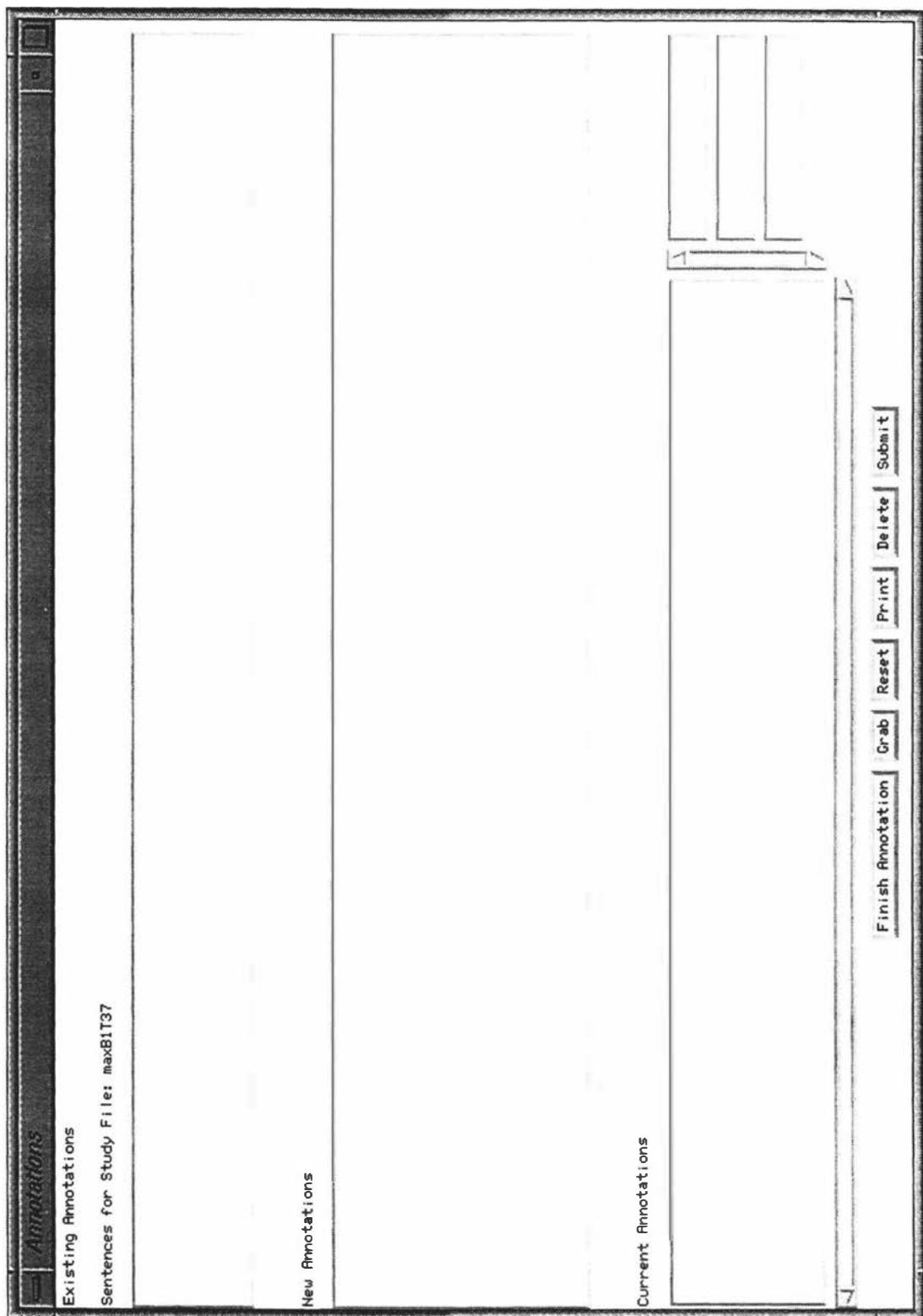
Sentences for Study File: maxB1T37

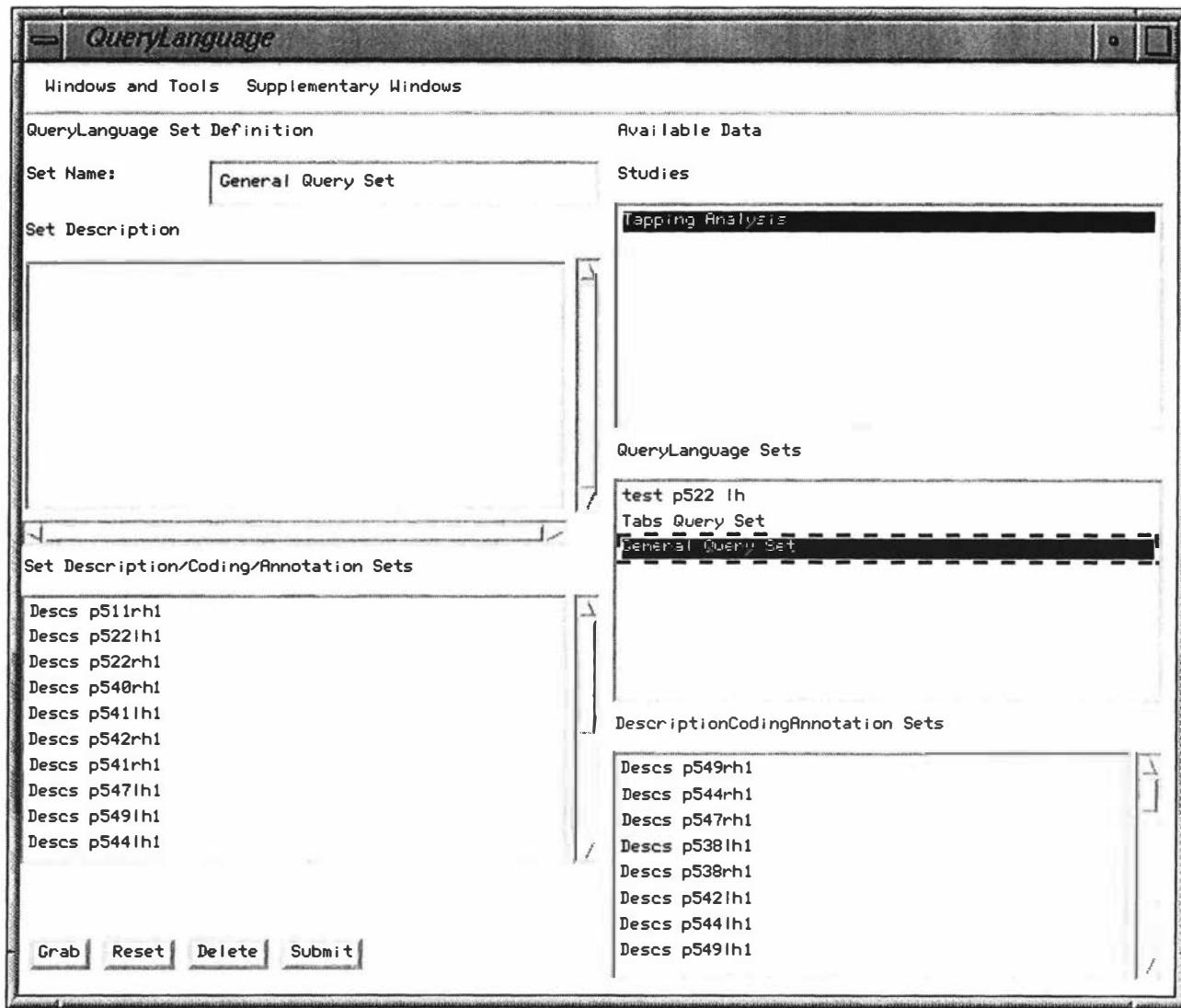
B1 DS experi obs1 while Max NonVocalisation	00:00:00.00	00:00:04.23
B1 DS experi obs1 while Max whining	00:00:04.23	00:00:45.81
B1 DS experi obs1 while Max NonVocalisation	00:00:45.81	00:00:49.49
B1 DS experi obs1 while Max whining	00:00:49.49	00:00:54.14
B1 DS experi obs1 while Max NonVocalisation	00:00:54.14	00:00:57.54
B1 DS experi obs1 while Max whining	00:00:57.54	00:01:04.38

New Descriptions/Codes

Current Description/Code

Finish Description/Coding Sentence Grab Reset new Sentence Text Reset new Sentences Reset Check
Print CalculatePeriods Delete Submit

Module Description: Window Annotations (Dog Calming Case Study)

Module QueryLanguage: Main Window (Parkinson's Disease Case Study)

APPENDIX

Module QueryLanguage: Window Formulate Query (Parkinson's Disease Case Study)

FormulateQuery

QueryLanguage Text

```
dyskinesia(tp) BEFORE hesitation WITHIN 300
```

Previous QueryLanguage Text

Constructors for Time or Position Comparisons:

Constructors for Time or Position Intervals:

Constructors for Sequences of Description/Coding Sentences:

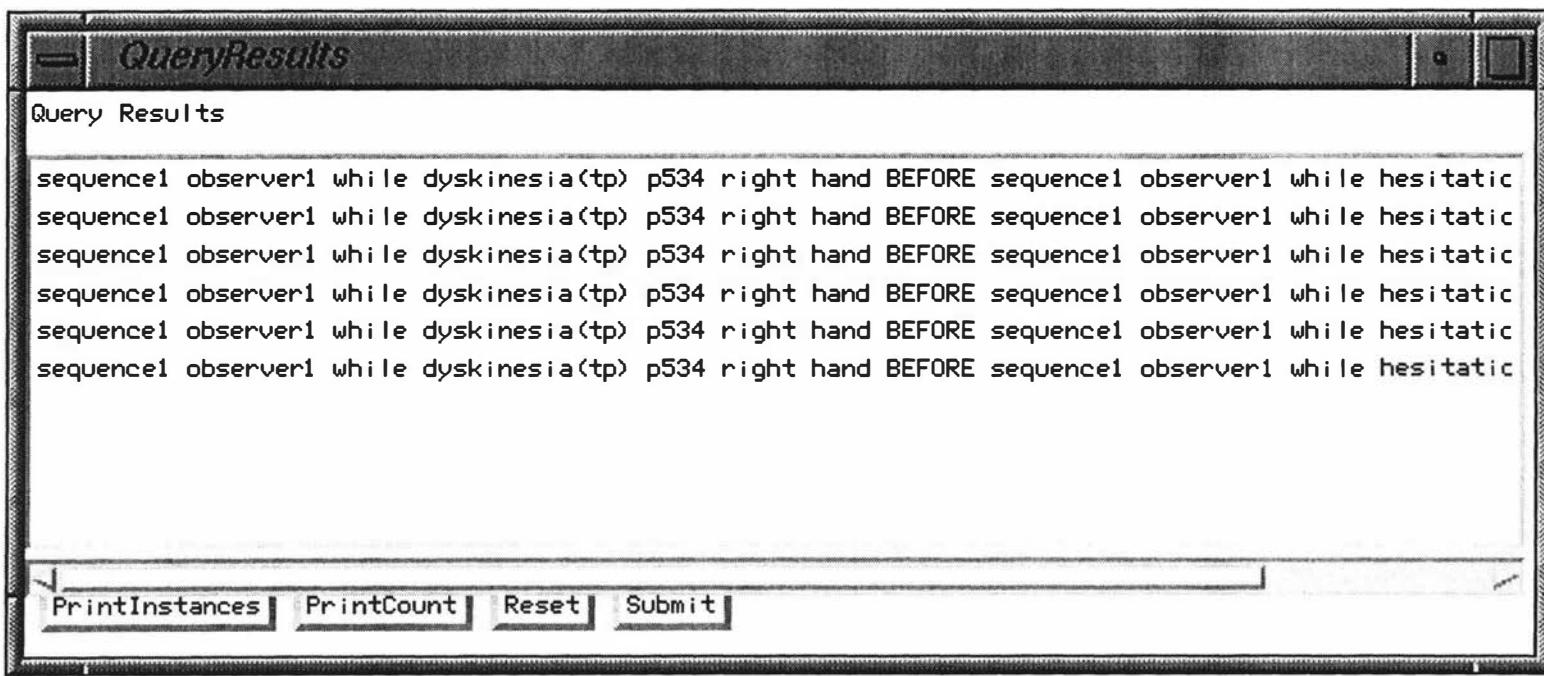
Constructors for Combination of Description/Coding Sentences:

Constructors for Boolean Comparisons of Description/Coding Sentences:

General Constructors:

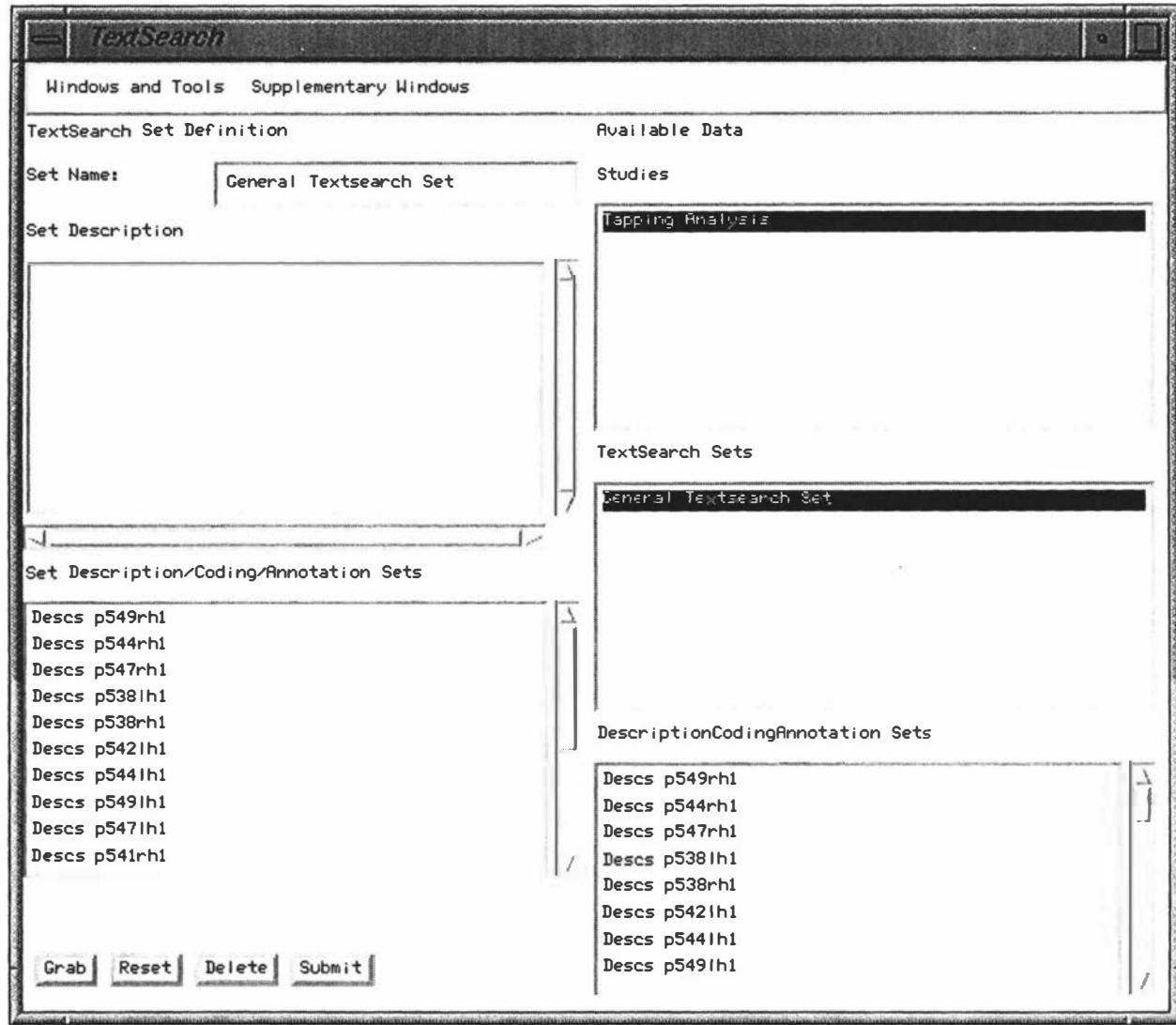
PlaceHolders:

Person/Thing	Action	Concept	Conjunction	Preposition	Descriptor
P5s p538 p547 p544 p549 p542 p541 p534	tapping	TappingConcepts tremor dyskinesia(hd) dyskinesia(tp) hesitation arrest amplitude frequency	while		highNormalLow high normal low leftRight right left higherLower



APPENDIX E

Module TextSearch: Main Window (Parkinson's Disease Case Study)



Module TextSearch: Windows Formulate Search and Search Results (Parkinson's Disease Case Study)

FormulateSearch

TextSearch Text

very low

Use Wildcards Case Sensitive

Reset TextSearch

SearchResults

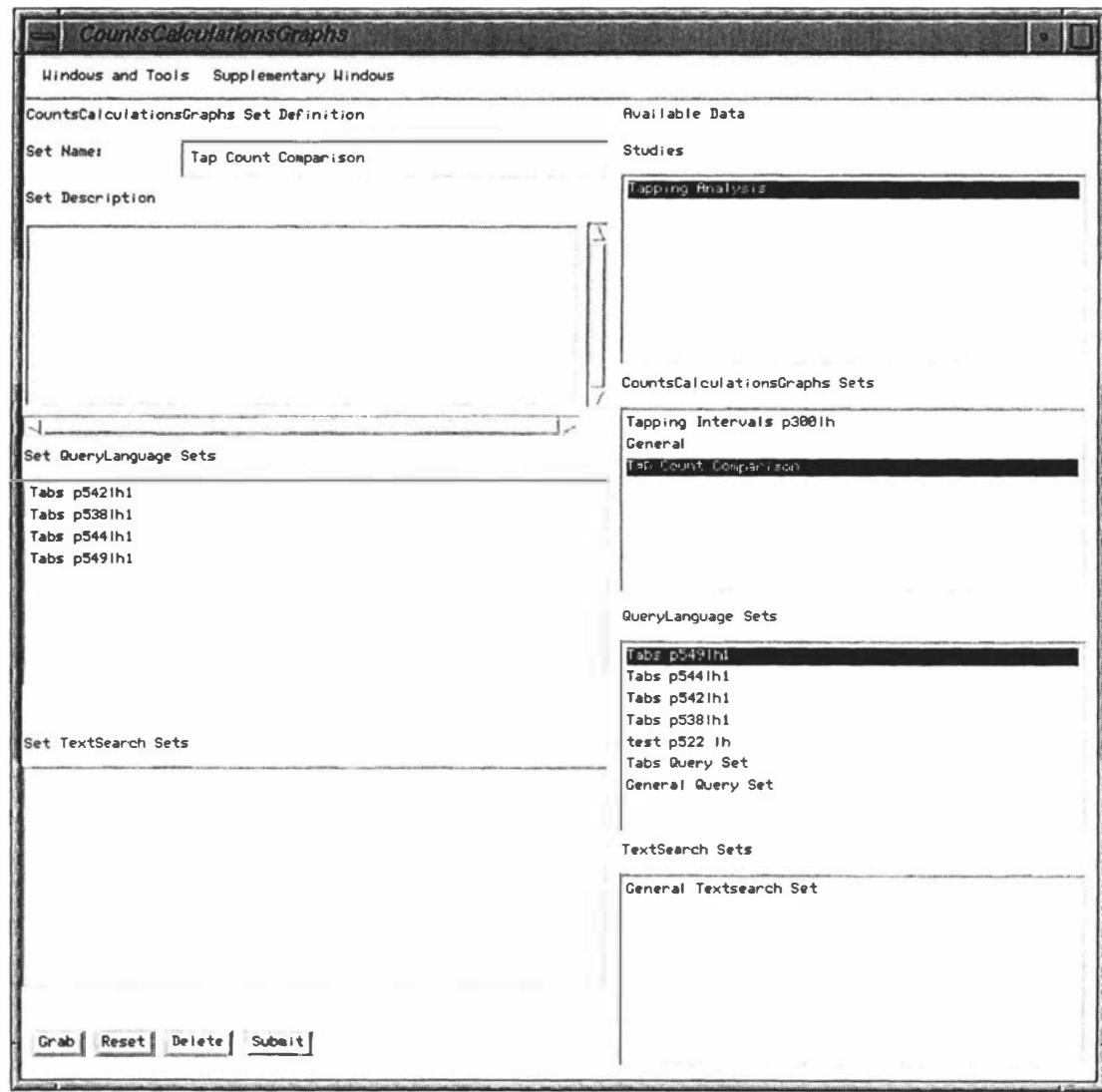
Text Search Results

High frequency and very low amplitude with tremor/hesitations and loss of contr	3000
High frequency and very low amplitude with continuous tremor/dyskinesia of the h	3000

Print Reset Submit

APPENDIX E

Module CountsCalculationsGraphs: Main Window (Parkinson's Disease Case Study)

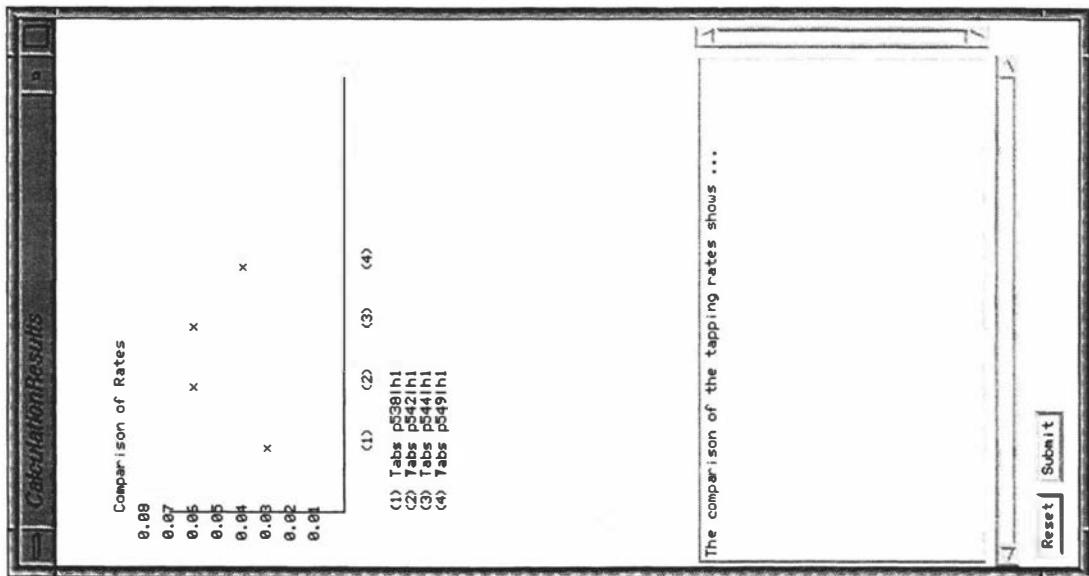


Module CountsCalculationsGraphs: Window Calculations (Parkinson's Disease Case Study)

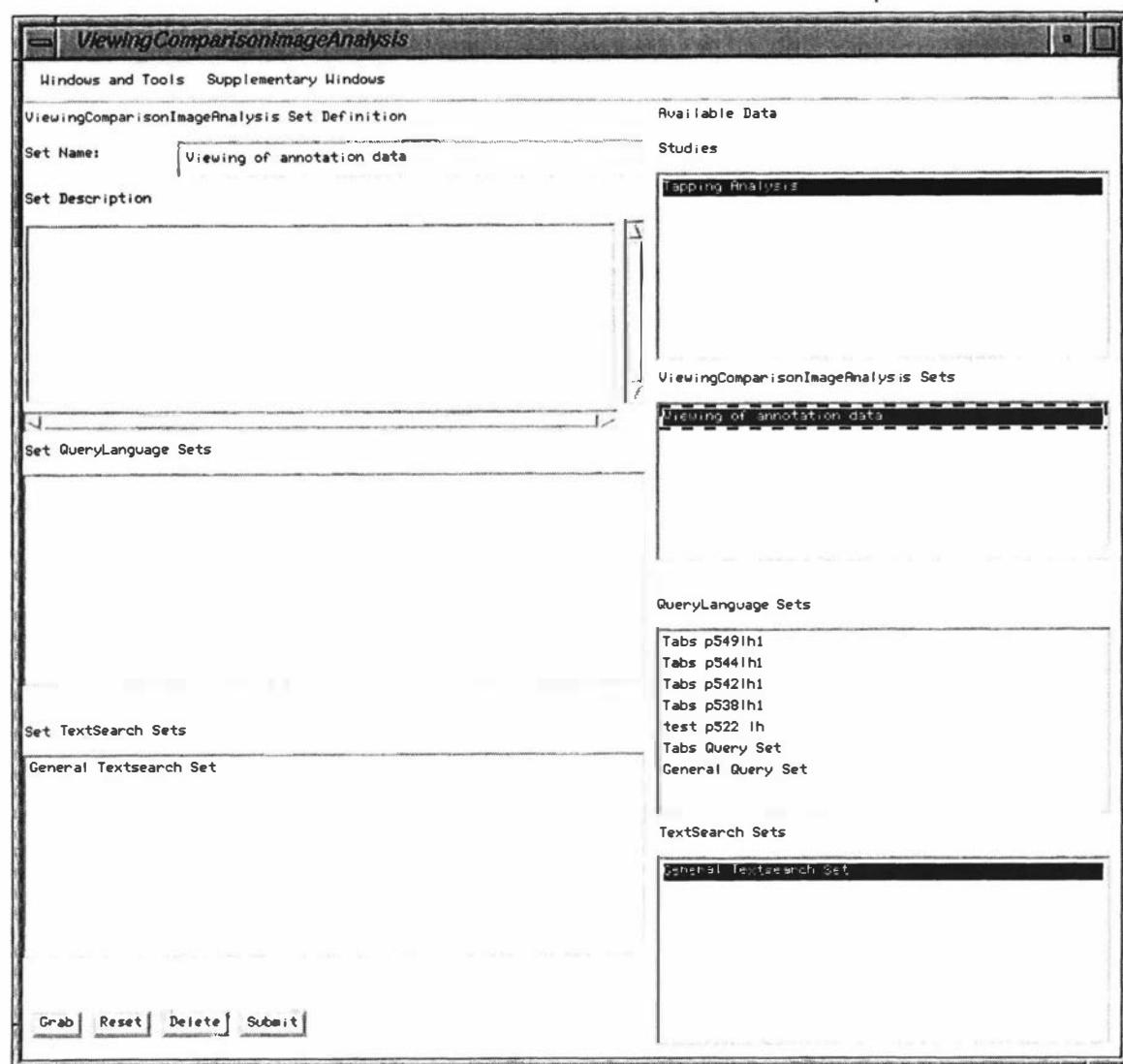
Calculations						
QueryLanguage Sets	Count	#StudyFiles	MediaTypes	TotalDuration	AverageDuration	Rate
Tabs p542lh1	108	1	V	1	0	0.06
Tabs p538lh1	65	1	V	0	0	0.03
Tabs p544lh1	109	1	V	0	0	0.06
Tabs p549lh1	72	1	V	0	0	0.04

TextSearch Sets	Count	#StudyFiles	MediaTypes	TotalDuration	AverageDuration	Rate

Sets selected for Calculation	Rate	Graph	Compare
Tabs p538lh1 Tabs p542lh1 Tabs p544lh1 Tabs p549lh1	<input type="button" value="Reset"/>	<input type="button" value="Graph"/>	<input type="button" value="Compare"/>

Module CountsCalculationsGraphs: Window Calculation Results (Parkinson's Disease Case Study)

Module ViewingComparisonImageAnalysis: Main Window (Parkinson's Disease Case Study)



**Module ViewingComparisonImageAnalysis: Window Viewing Comparison
(Parkinson's Disease Case Study)**

ViewingComparison

(1)General Textsearch Set	p522lh1	3000	3000 Normal tapping with slight loss of control near the end.
(2)General Textsearch Set	p522rhl	3000	3000 Normal tapping with slight loss of control after 20 s
(3)General Textsearch Set	p534lh1	3000	3000 Normal frequency, but low amplitude and some inconsistency in both
(4)General Textsearch Set	p542rhl	3000	3000 Normal frequency and amplitude, but with slight loss of control
(5)General Textsearch Set	p547lh1	3000	3000 Normal frequency and low amplitude with infrequent hesitations.
(6)General Textsearch Set	p549lh1	3000	3000 Normal frequency and just normal amplitude with some hesitations at
(7)General Textsearch Set	p544lh1	3000	3000 High frequency and low amplitude (combined with tremor/hesitations)
(8)General Textsearch Set	p542lh1	3000	3000 High frequency and very low amplitude with continuous tremor/dyskinesia
(9)General Textsearch Set	p547rhl	3000	3000 Normal frequency and just normal amplitude with no hesitations
(10)General Textsearch Set	p544rhl	3000	3000 High frequency and very low amplitude with tremor/hesitations and
(11)General Textsearch Set	p549rhl	3000	3000 Normal frequency and just normal amplitude with no hesitations but

Display 1 Display 2 Display 3 Display 4

(1)Normal tapping with slight loss c
(7)High frequency and low amplitude

(2)Normal tapping with slight loss c
(10)High frequency and very low ampli

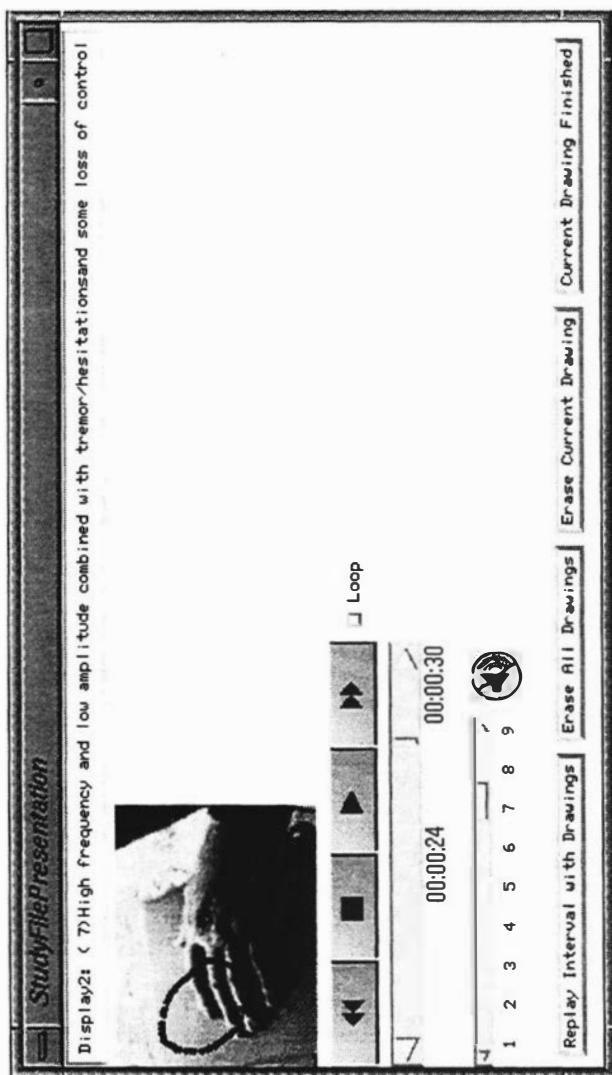
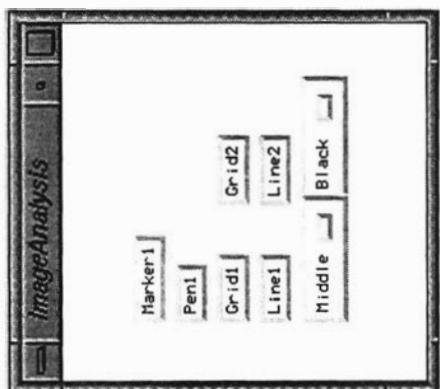
Grab Remove Reset
Leadin Leadout

200	0
-----	---

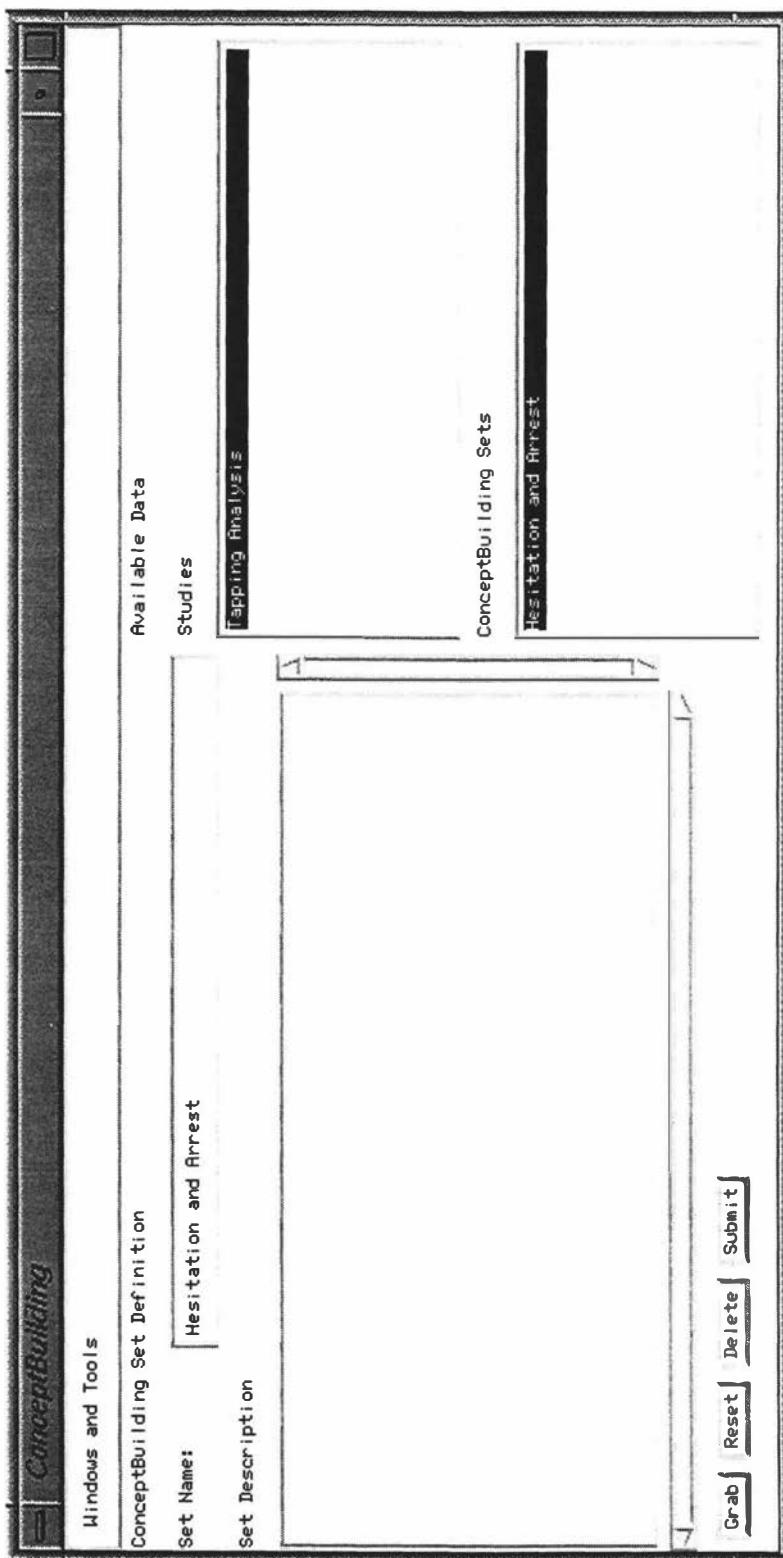
Controls val

First Previous Same Next Last

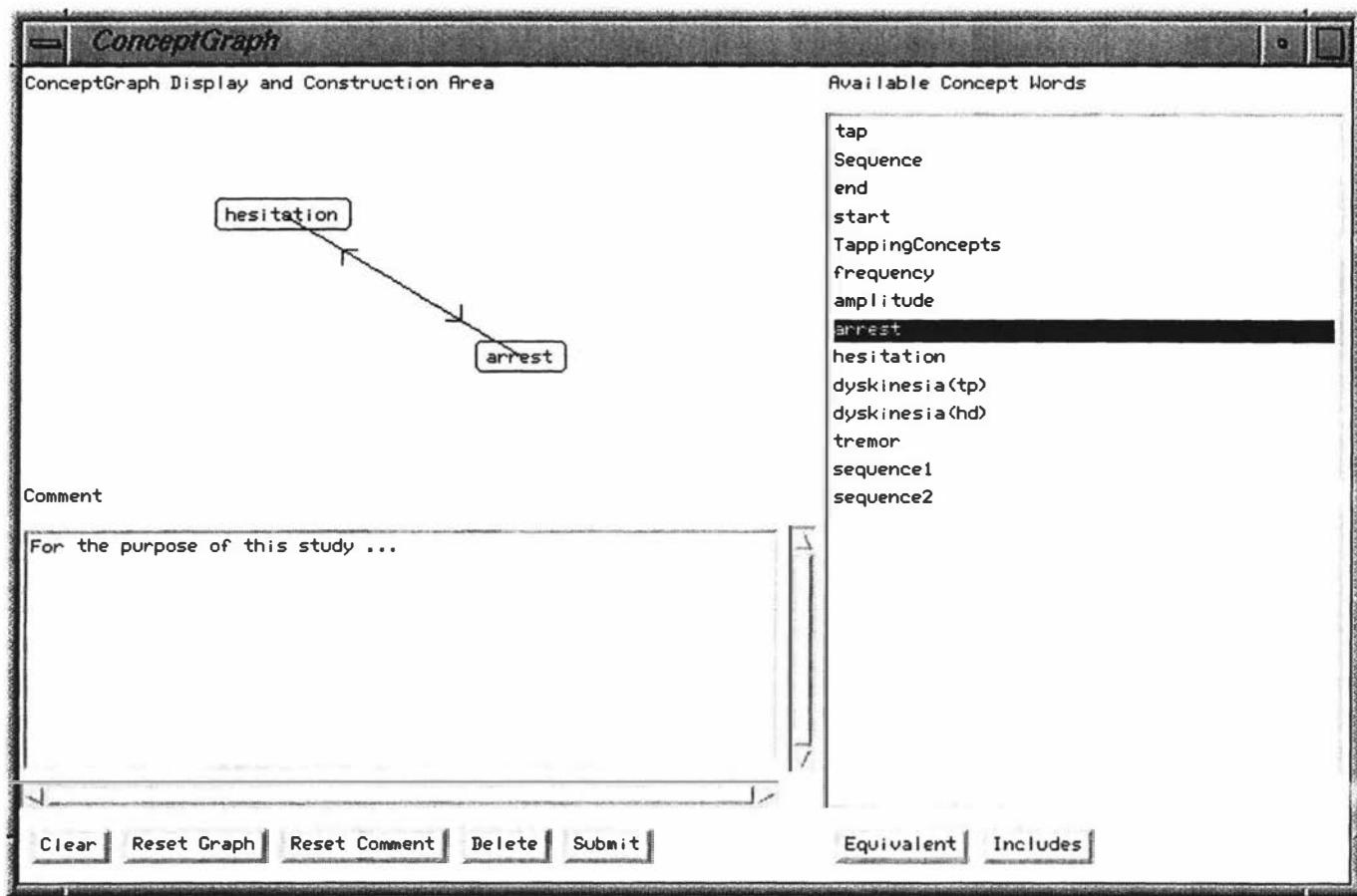
Reset Submit

Module ViewingComparisonImageAnalysis: Window Image Analysis (Parkinson's Disease Case Study)

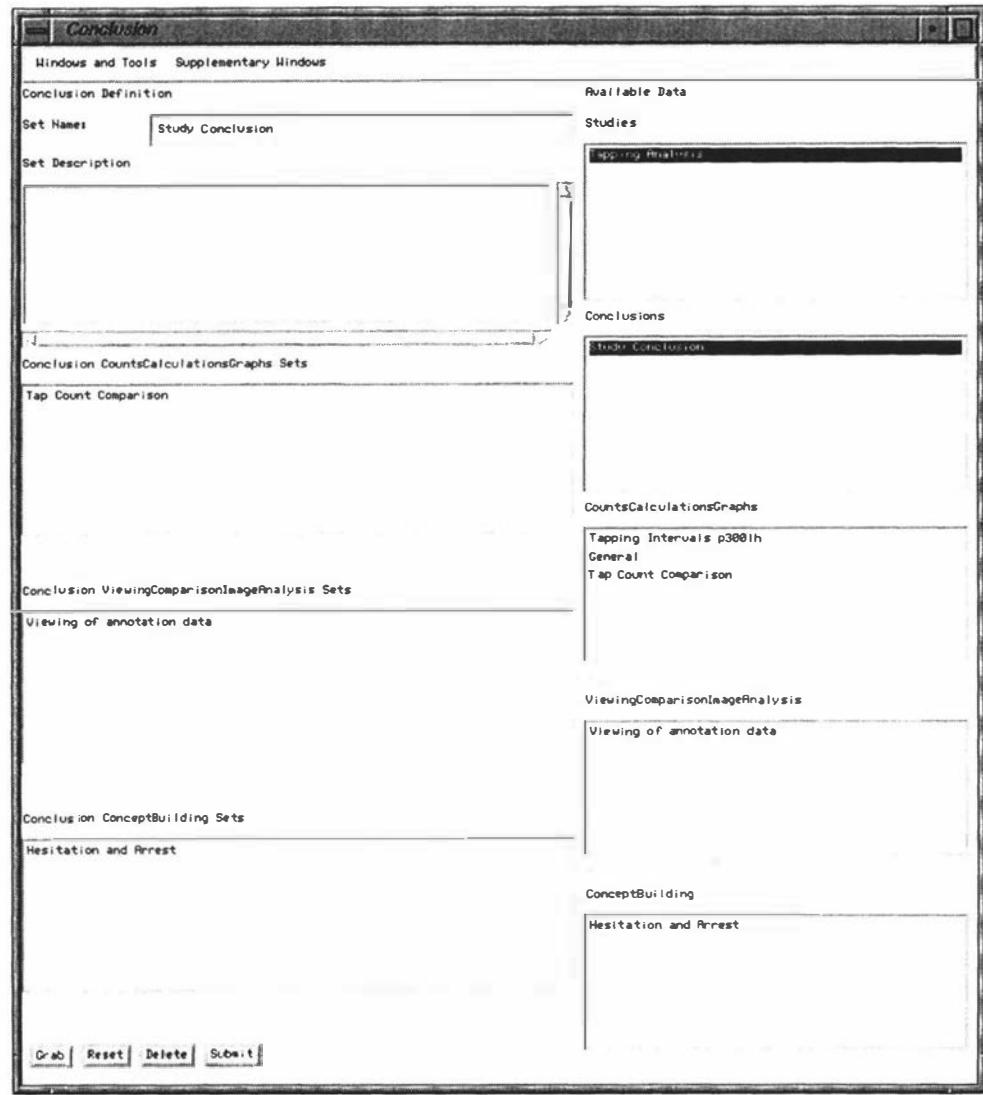
Module ConceptBuilding: Main Window (Parkinson's Disease Case Study)



Module ConceptBuilding: Window Concept Graph (Parkinson's Disease Case Study)



Module Conclusion: Main Window (Parkinson's Disease Case Study)

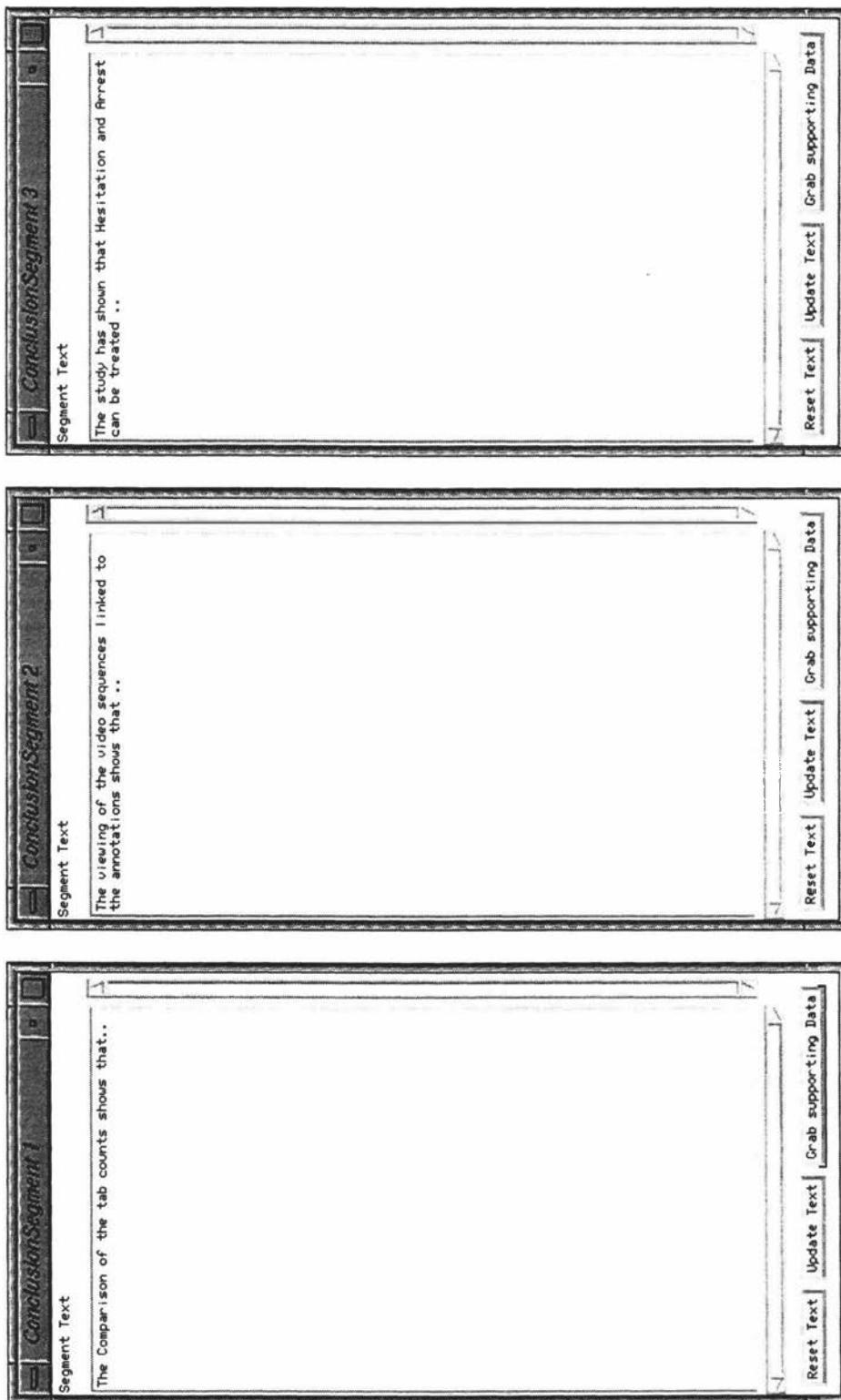


Module Conclusion: Window Conclusion Construction (Parkinson's Disease Case Study)

ConclusionConstruction

Start of Conclusion Statement	Data Type	Data Name
1	CCG	Tap Count Comparison
2	VCI	Viewing of annotation data
3	CB	Hesitation and Arrest
new		

Reset | MoveSegmentDown | MoveSegmentUp | RemoveSegment | Delete | Submit |

Module Conclusion: Windows Conclusion Segments (Parkinson's Disease Case Study)

Module Conclusion: Window Conclusion Replay and Data (Parkinson's Disease Case Study)