

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.

**THE
USE OF
FRAMES
IN
KNOWLEDGE-BASED
SYSTEMS**

A thesis
submitted in partial fulfilment
of the requirements for the Degree of
Master of Science in Computer Science
at Massey University by

WILLIAM JOHN TEAHAN

Massey University
1986

ABSTRACT

The general aim of this study was to investigate the use of frames as a means of representing knowledge in computer knowledge-based systems. This thesis examines the application of frames to two particular situations, the playing of an opening bid in Bridge, and the recognition of birds from field observations. The Frame Representation Language FRL was used in the implementation of the two different systems.

Three aspects of frames are investigated : the problems of matching two different frames; the problems of structuring frame systems for searching; and the problem of improving the interface between the frame system and the user of the knowledge base. A comparison is also made of frames with other methods of knowledge representation such as production systems and semantic networks. Finally, further areas of research into the use of frames are suggested such as the extension of frame matching, research into the aspects of knowledge representation and application of frames to specific problems.

ACKNOWLEDGEMENTS

Thanks to my supervisor Ray Kemp for providing help and suggestions throughout the preparation of this thesis.

CONTENTS

	Page
1 Introduction	1
2 Matching Frames	6
2.1 Problems with Matching Frames	6
2.2 Use of \$REQUIRE Facets in Generic Frames	11
2.3 Defining an Alternative Matching Scheme	13
2.4 Using \$MATCH Facets to Specify the Matching Function	15
2.5 Improving the Matching Structure Using \$IF-MATCHED Demons	19
2.6 Using the MATCH Slot to Specify the Frame Match	21
3 Structuring Frames for Search	24
3.1 Different Frame Structures	24
3.2 Searching Strategies	25
3.3 Linear Structure	27
3.4 Set Structure	29
3.5 Hierarchical Structure	34
3.5.1 Search Tree	37
3.6 Network Structures	41
3.6.1 Slot Network	42
3.6.2 AKO Network	47
3.6.3 Further Network Structures	56
3.7 Analysis of the Different Frame Structures	56
4 Improving the User Interface	60
4.1 Problems with Interfacing Between the Frame System and the User	60
4.2 Improving the Presentation by Using \$ENTER and \$DISPLAY Facets	61

4.3	Improving the User Search Interface	69
4.3.1	Specifying the Search Using the FENTER command	69
4.3.2	Using \$IF_MATCHED Demons to Provide a Trace of the Search	70
4.3.3	Using Interactive Matching Functions	70
4.4	Allowing the User to Modify the Knowledge Base	76
4.4.1	Addition of Frames to the Knowledge Base	76
4.4.2	Removal of Frames from the Knowledge Base	79
4.4.3	Alteration of a Frame in the Knowledge Base	80
4.4.4	The Effect of Modifying the Knowledge Base on the Efficiency of the Search	82
4.5	Allowing the Expert to Create the Knowledge Base	83
4.5.1	Using Lists of Pre-defined Functions to Specify the Procedural Information	84
4.5.2	Interrogating the User about the Procedural Information	86
4.5.3	Problems with Allowing the Expert to Create the Knowledge Base	86
5	Comparison of Knowledge Representation Methods	89
5.1	Features of Knowledge Representations	89
5.2	Methods of Representing Knowledge	91
5.2.1	Production Systems	92
5.2.2	Semantic Networks and Property Lists	97
5.2.3	Frames	100
6	Further Lines of Research	104
6.1	Extensions to Frame Matching	104
6.1.1	Approximate Frame Matches	104
6.1.2	Using Ideas to Match the Frames	106
6.1.3	Matching the AKO Slot	107
6.1.4	Dynamic Matching Schemes	108
6.1.5	Using Matching to Perform the Search	110
6.2	Research into Aspects of Knowledge Representation	110
6.3	Application of Frames to Specific Problems	111
7	Summary	113

Appendix A. FRL Commands	A-1
Descriptions	A-1
FDISPLAY	A-5
FENTER	A-6
FMATCH	A-7
FSEARCH	A-12
FSEARCH-AKO-NETWORK	A-13
FSEARCH-SET	A-15
FSEARCH-SLOT-NETWORK	A-16
FSEARCH-TREE	A-17
FVALUES	A-18
 Appendix B. Frame Definitions	 B-1
B-1 Bridge System	B-1
B-1-1 Definition of HAND	B-1
B-1-2 Sample Attached Functions	B-2
B-1-3 Sample Linear Frames	B-6
B-1-4 Sample Set Frames	B-7
B-1-5 Sample Hierarchical Frames	B-8
B-1-6 Sample Search Tree Frames	B-10
B-1-7 Sample Slot Network Frames	B-11
B-1-8 Sample AKO Network Frames	B-12
B-2 Bird Recognition System	B-13
B-2-1 Definition of BIRD	B-13
B-2-2 Sample Attached Functions	B-16
B-2-3 Sample Linear Frames	B-20
B-2-4 Sample Set Frames	B-22
B-2-5 Sample Search Tree Frames	B-24
B-2-6 Sample AKO Network Frames	B-26

Appendix C. Trace of Bridge System	C-1
C-1 Presentation to the User	C-1
C-2 Sample Bridge Hands	C-3
Appendix D. Trace of Bird Recognition System	D-1
D-1 Presentation of the Birds	D-1
D-2 Specific Search	D-6
D-3 General Query	D-19
Bibliography	

1 INTRODUCTION.

An area of increasing importance in Artificial Intelligence in recent years has been the problem of how to represent knowledge on the computer. Some examples of methods that have been developed are production systems and frames. This thesis investigates the features of frames in particular and how they can be used to construct knowledge bases which can be incorporated into expert systems.

Marvin Minsky, in "A Framework for Representing Knowledge", first proposed the theory of frames in 1974. He defined a frame as being "a data-structure for representing a stereotyped situation." Some examples of such situations are entering a room, driving to work and watching television. The frame representation language FRL was developed in 1977 by Goldstein and Roberts to implement the theory of frames. The language is mostly declarative in that it depends on data structures as opposed to procedures for the definition of the frames.

Each frame consists of various types of information described by slots. Each slot contains any number of facets which define how the information in the slot is to be used. Each facet in turn consists of values which contain the actual data or information that is being represented. Attached to each data item may also be several comments. Each comment consists of a label and a message.

Related frames are organised into frame systems. Each frame of the system share the same slots so that the same functions can be applied to all the frames in the system. The frame systems can also be structured into information retrieval networks which provide alternative frames to search when a frame fails to match a particular

situation. Diagram 1.1 illustrates the hierarchical structure of a frame system.

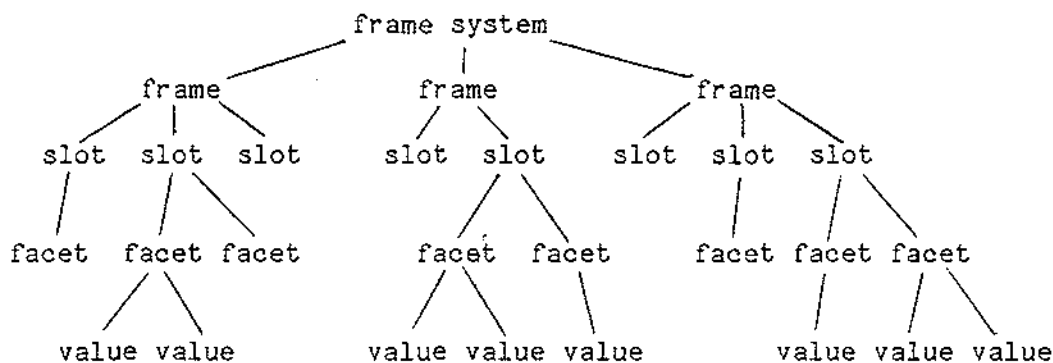


Diagram 1.1 Structure of a frame system.

Three basic instructions, FGET, FPUT and FREMOVE describe the main operations for obtaining, inserting and removing information stored in the frames. Four important features of frames are defaults, demons, inheritance and requirements.

Each slot in a frame can have a value facet which describes the data values associated with each slot. Alternatively, the slot can have a default facet which is used when there is no value facet exists. This allows for general assumptions about the information to be stored in the frames which can then be displaced at a later date when more specific data arrives that better fits the current situation.

Functions that are activated automatically when a specific situation occurs are called demons. The demons are expressed as functions attached to various facets in each slot of a frame. Examples of demons used in frames are if-added, if-removed and if-needed demons which are activated whenever the information in the slot is added to, removed or needed. These demons can activate further demons and hence a simple change or reference to a frame can initiate a whole series of

actions that may affect other frames in the system.

Another powerful feature of frames is the ability to use information from other frames through inheritance. Related frames in the system can be linked through an AKO (A-KIND-OF) slot which indicate that a particular frame has similar properties to the related frame. This means that information can be 'inherited' through the pathway and does not need to be stored in the frame itself.

Values within a slot may be restricted by certain requirements that describe the allowable values for the slot. Frames in an AKO hierarchy can therefore be classified as being generic, where general requirements are used to describe the frame, or individual, where more specific values are used.

The particular implementation of FRL used throughout this thesis was developed at MIT on the PRIME 750 computer and is incorporated into the V-mode LISP language available on the computer.

Two particular applications of frames have been investigated and are referred to throughout this work. These are :

1. finding an opening bid in bridge;
2. recognition of birds from field observations.

The first application relates to the problem of finding an appropriate opening bid in bridge such as 1 Spade or 2 Clubs given the cards in the player's hand. The bidding system used in the implementation is Acol. The second application involves the

recognition of birds from field observations of characteristics such as habitat and appearance. The set of birds used for this application is arbitrarily limited to the common town and pasture birds in New Zealand. The names of the frame systems developed for these two applications are, respectively, the Bridge System and the Bird Recognition System. The choice of the two different applications are sufficiently diverse to examine the versatility of frames when applied to different kinds of information.

The work described in the following chapters falls into three main areas of investigation :

1. matching two different frames;
2. structuring frame systems for search;
3. improving the interface between the knowledge base and the user.

The first area deals with the problem of comparing two separate frames with different information to see if they match. The use of requirements to express a generic frame against which an individual frame is matched is described. An alternative matching scheme is also proposed which uses matching functions attached to a generic frame to define how the frame is to be matched. The need for matching demons and a method to express how a frame is to be matched is also described.

The second area of research is the problem of structuring frame systems for search to find the particular frame or frames that match a given frame. This involves ordering the frame system in some manner or linking the frames in the system by reference to other frames. Types of structures investigated are : linear, set, hierarchical and network

structures.

The third area looks at how the frame systems can be organised so that the interface with the user can be improved. For improving the presentation of frames to the user, the use of attached functions to display and enter values in a frame is proposed. Various methods of improving the search of the knowledge base for the user are described, such as using matching demons for tracing and interactive match functions for querying the user. The problems associated with allowing the user to create or modify the frame systems are also examined.

After the description of these three areas of research, a comparison is made of frame-type structures with other systems of knowledge representation such as decision-trees, production systems and procedures. Following this is a description of improvements, current developments and useful lines of investigation on frame systems in general. Various further applications of the frame systems to other areas of knowledge are also explored.