

October 1981

Report. No. STAN-CS-81-885

*Also numbered:  
HPP-81-16*

# The Emycin Manual

by

W. Van Melle

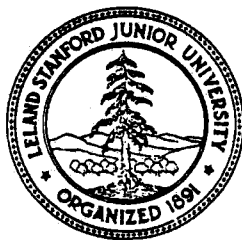
A. C. Scott

J. S. Bennett

M. Peairs

**Department of Computer Science**

Stanford University  
Stanford, CA 94305



# The EMYCIN Manual

William van Melle, A. Carlisle Scott  
James S. Bennett, and Mark A. S. Peairs

9 November 1981

## Abstract

This manual describes a domain-independent system, called EMYCIN, for constructing one class of expert computer programs: rule-based consultants. The resulting programs use knowledge specific to a problem domain to provide consultative advice to a client. The system-building tool, EMYCIN, is based on the domain-independent core of the MYCIN program. Domain knowledge is represented in EMYCIN systems primarily as production rules, which are applied by a goal-directed backward-chaining control structure. Rules and consultation data may have associated measures of certainty, and incomplete data entry is allowed. The system includes an explanation facility that can display the line of reasoning followed by the consultation program, and answer questions from the client about the contents of its knowledge base.

To aid the system designer in producing a knowledge base for a domain quickly and accurately, EMYCIN provides the following features: (1) a terse, stylized, but easily understood language for writing rules; (2) extensive checks to catch common user errors, such as misspellings; and (3) methods for handling all necessary bookkeeping chores. Other human-engineering features allow the system architect to produce, with a minimum of effort, a consultation program that is pleasing in appearance to the client.

Several consultation programs have been developed using EMYCIN, including consultants for medical problems and a consultant for structural analysis.

Copyright (c) 1981 The Board of Trustees of Leland Stanford Junior  
University. All Rights Reserved.

Research supported in part by NSF grant MCS-7903753

## Table of Contents

<b>1. Introduction</b>	<b>3</b>
<b>2. Contexts</b>	<b>15</b>
2.1. The Context Tree	15
2.2. Uses of the Context Tree	18
2.3. Defining Contexts	21
2.4. Parameter Groups and Rule Groups	22
2.5. Properties of Context Types	22
<b>3. Parameters</b>	<b>27</b>
3.1. Defining Parameters	30
3.2. Properties of Parameters	31
3.3. Examples of Parameters	36
<b>4. Rules</b>	<b>39</b>
4.1. The Structure and Definition of Rules	39
4.2. The Rule Interpreter	41
4.3. Rule Groups	44
4.4. The Uses of Rules	46
4.5. Abbreviated Rule Language (ARL)	51
<b>5. Building an EMYCIN Consultant</b>	<b>55</b>
5.1. Organizing your Knowledge Base	55
5.2. Example: Acquiring a Business Consultant	57
<b>6. Creating and Maintaining the Knowledge Base</b>	<b>63</b>
6.1. Putting the Knowledge Base On-line	63
6.2. The EMYCIN Executive	64
6.3. The Knowledge Base Editors	65
6.4. Saving the Knowledge Base	66
6.5. Listing the Knowledge Base	71
<b>7. Debugging the Knowledge Base</b>	<b>75</b>
7.1. Consultation Tracing	75
7.2. WHY and HOW Questions	75
7.3. The EXPLAIN, TEST, and REVIEW Options	77
7.4. The QA Module	81
7.5. Debugging with Batch	84
<b>8. Consultation Options</b>	<b>89</b>
8.1. Consultation Special Options	89
8.2. Maintaining Consultant Cases	92
8.3. Controlling the Interaction of the Consultant	93
<b>9. Implementation Details</b>	<b>101</b>
9.1. Miscellaneous System Variables	101
9.2. Pointers to and from Rules	101

9.3. Internal Data Structures during a Consultation	102
9.4. Modifying QA for your domain	104
9.5. The Dictionary	105
9.6. Defining New Predicate and Action Functions	107
<b>Appendix I. Functions</b>	<b>111</b>
I.1. Non-Numeric Predicate Functions	111
I.2. Numeric Predicate Functions	114
I.3. Conclusion Functions	116
I.4. Auxiliary functions	118
I.5. Functions for TEXT-valued Parameters	119
I.6. Mapping functions	121
<b>Appendix II. An Introduction to the Interlisp editor</b>	<b>127</b>
<b>Index</b>	<b>135</b>

## List of Figures

Figure 1-1: The major roles of EMYCIN	4
Figure 1-2: A sample rule from the domain of structural analysis	6
Figure 1-3: Example of ARL format for rule input/output.	10
Figure 2-1: SACON's static tree of context types	16
Figure 2-2: An instance tree from SACON	16
Figure 2-3: MYCIN's static tree	18
Figure 2-4: A tree of instances during a particular MYCIN consultation	19
Figure 2-5: LITHO's static tree and an instance tree	19
Figure 2-6: CLOT's static tree and an instance tree	20
Figure 2-7: GRAVIDA's static tree and instance tree	20
Figure 2-8: An example of a dynamic relationship between context instances	21
Figure 2-9: Prompt properties required by a context	24
Figure 3-1: Parameter Editor commands	30
Figure 4-1: A sample rule from the CLOT system.	39
Figure 4-2: CFCOMBINE	44
Figure 4-3: MYCIN's static tree	45
Figure 4-4: ARL Operators	53
Figure 4-5: ARL Forms	53
Figure 5-1: SACON Inference Structure	56
Figure 5-2: A Goal Rule for SACON	57
Figure 6-1: EMYCIN Executive commands	64
Figure 6-2: Property Editor commands	66
Figure 6-3: Migration of Changes	68
Figure 6-4: Knowledge base files for a consultant	70
Figure 7-1: QA Options	82
Figure 7-2: Legal question forms for QA	83
Figure 8-1: Special Options Available Before a Consultation	89
Figure 8-2: Instructions for Running Consultations	90
Figure 8-3: Possible Answers During a Consultation	91
Figure 8-4: Special Options during TAB	95
Figure 9-1: Non-Numeric Predicate Functions	112
Figure 9-2: Interlisp Editor Commands	131

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.