

Eta S. Berner *Editor*

Clinical Decision Support Systems

Theory and Practice



Second Edition

HEALTH INFORMATICS SERIES

Health Informatics

(formerly Computers in Health Care)

Kathryn J. Hannah Marion J. Ball
Series Editors

Eta S. Berner
Editor

Clinical Decision Support Systems

Theory and Practice

Second Edition

 Springer

Eta S. Berner, EdD, FACMI, FHIMSS
Professor
Health Informatics Program
Department of Health Services Administration
School of Health Professions
University of Alabama at Birmingham
Birmingham, AL, 35294
USA

Series Editors:

Kathryn J. Hannah, PhD, RN
Adjunct Professor
Department of Community
Health Science
Faculty of Medicine
The University of Calgary
Calgary, Alberta T2N 4N1
Canada

Marion J. Ball, EdD
Professor, Johns Hopkins
School of Nursing
Joint Appointment
Health Sciences Informatics
Johns Hopkins University
School of Nursing
and
Affiliate Professor
Information Systems
University of Maryland
Baltimore
and
Fellow
Global Leadership Initiative
Center for Healthcare Management
Baltimore, MD 21210
USA

Library of Congress Control Number: 2006924586

ISBN-10: 0-387-33914-0

e-ISBN-10: 0387-38319-0

ISBN-13: 978-0387-33914-6

e-ISBN-13: 978-0-387-38319-4

Printed on acid-free paper.

© 2007, 1999 Springer Science+Business Media, LLC

All rights reserved. This work may not be translated or copied in whole or in part without the written permission of the publisher (Springer Science+Business Media, LLC, 233 Spring Street, New York, NY 10013, USA), except for brief excerpts in connection with reviews or scholarly analysis. Use in connection with any form of information storage and retrieval, electronic adaptation, computer software, or by similar or dissimilar methodology now known or hereafter developed is forbidden.

The use in this publication of trade names, trademarks, service marks, and similar terms, even if they are not identified as such, is not to be taken as an expression of opinion as to whether or not they are subject to proprietary rights.

While the advice and information in this book are believed to be true and accurate at the date of going to press, neither the authors nor the editors nor the publisher can accept any legal responsibility for any errors or omissions that may be made. The publisher makes no warranty, express or implied, with respect to the material contained herein.

9 8 7 6 5 4 3 2 1

springer.com

Series Preface

This series is directed to healthcare professionals who are leading the transformation of health care by using information and knowledge. Launched in 1988 as *Computers in Health Care*, the series offers a broad range of titles: some addressed to specific professions such as nursing, medicine, and health administration; others to special areas of practice such as trauma and radiology. Still other books in the series focus on interdisciplinary issues, such as the computer-based patient record, electronic health records, and networked healthcare systems.

Renamed *Health Informatics* in 1998, to reflect the rapid evolution in the discipline now known as health informatics, the series will continue to add titles that contribute to the evolution of the field. In the series, eminent experts, serving as editors or authors, offer their accounts of innovations in health informatics. Increasingly these accounts go beyond hardware and software to address the role of information in influencing the transformation of healthcare delivery systems around the world. The series also will increasingly focus on “peopleware” and organizational, behavioral, and societal changes that accompany the diffusion of information technology in health services environments.

These changes will shape health services in the new millennium. By making full and creative use of the technology to tame data and to transform information, health informatics will foster the development of the knowledge age in health care. As coeditors, we pledge to support our professional colleagues and the series readers as they share advances in the emerging and exciting field of health informatics.

Kathryn J. Hannah, PhD, RN
Marion J. Ball, EdD

Preface

When the first edition of *Clinical Decision Support Systems* was published in 1999, I began the preface with the statement, “We are at the beginning of a new era in the application of computer-based decision support for medicine.” Usually such statements in hindsight seem unduly optimistic, but if we look at the landscape of healthcare information technology today, that assessment appears to be surprisingly accurate. Shortly after the book was published, the first of several landmark reports from the Institute of Medicine on the quality of health care led to greater awareness of the role these systems can play in improving patient safety and healthcare quality. This second edition is being published at a time when there is increased governmental, research and commercial interest in clinical decision support systems (CDSS). The purpose of this book is to provide an overview of the background and state-of-the-art of CDSS. A persistent theme is that CDSS have enormous potential to transform health care, but developers, evaluators, and users of these tools need to be aware of the design and implementation issues that must be addressed for that potential to be realized as these systems continue to evolve.

This book is designed to be (1) a resource on clinical decision support systems for informatics specialists; (2) a textbook for teachers or students in health or medical informatics training programs; and (3) a comprehensive introduction for clinicians, with or without expertise in the applications of computers in medicine, who are interested in learning about current developments in computer-based clinical decision support systems.

The book includes chapters by nationally recognized experts on the design, evaluation and application of these systems. This edition includes updates of chapters in the first edition, as well as several entirely new chapters. Section I provides background on CDSS development and evaluation. The first chapter introduces the topics that are explored in depth in later chapters. Chapters 2 through 4 describe the design foundations behind the decision support tools used today. While there is some overlap in the concepts addressed in each of these chapters, they each have unique foci. Chapter 2 focuses primarily on the mathematical foundations of the

knowledge-based systems, Chapter 3 focuses on pattern recognition systems, and Chapter 4 includes a detailed discussion of issues in clinical vocabularies and other important issues in the development and use of CDSS. Chapter 5 addresses diagnostic decision support systems and sets this development in the context of the process of physician, not just computer, diagnosis. Chapters 6 and 7 address concerns in the deployment and evaluation of any computer application in health care. These issues include examining the legal, ethical, and evaluation issues that must be addressed when these systems are actively used in health care.

Section II focuses on the applications of these systems in clinical practice. This section includes three chapters from institutions that not only have a strong history of deployment of these systems, but also have performed the research and evaluation studies that provide perspective for others who are considering the use of these tools. The final chapter in this section provides guidance on the use of decision support tools for patients.

This book represents an effort, not just by the editor or the individual chapter authors, but by many others who have provided assistance to them. We are grateful for the support and encouragement of our editors at Springer and the assistance of Joy Ptacek and Muzna Mirza in the preparation of this manuscript. The National Library of Medicine and the Agency for Healthcare Research and Quality have provided much appreciated support for my own and many of the chapter authors' research on CDSS. Finally, I want to express my gratitude to my friend and colleague, C. Michael Brooks, Ed.D., who provided the initial and ongoing encouragement for my research activities in clinical decision support systems over the past twenty years.

Eta S. Berner, EdD, FACMI, FHIMSS

Contents

Series Preface v

Preface vii

Contributors xi

Section 1 Development and Evaluation of Clinical Decision Support Systems

- CHAPTER 1 Overview of Clinical Decision Support Systems 3
Eta S. Berner and Tonya J. La Lande
- CHAPTER 2 Mathematical Foundations of Decision Support Systems 23
S. Andrew Spooner
- CHAPTER 3 Data Mining and Clinical Decision Support Systems 44
J. Michael Hardin and David C. Chhieng
- CHAPTER 4 Design and Implementation Issues 64
Jerome H. Carter
- CHAPTER 5 Diagnostic Decision Support Systems 99
Randolph A. Miller and Antoine Geissbuhler
- CHAPTER 6 Ethical and Legal Issues in Decision Support 126
Kenneth W. Goodman
- CHAPTER 7 Clinical Trials of Information Interventions 140
E. Andrew Balas and Suzanne Austin Boren

Section 2 Applications of Clinical Decision Support Systems in Clinical Practice

- CHAPTER 8 Clinical Decision Support at Intermountain Healthcare 159
Peter J. Haug, Reed M. Gardner, R. Scott Evans, Beatriz H. Rocha, and Roberto A. Rocha
- CHAPTER 9 Clinical Decision Support Within the Regenstrief Medical Record System 190
Burke W. Mamlin, J. Marc Overhage, William Tierney, Paul Dexter, and Clement J. McDonald
- CHAPTER 10 Decision Support During Inpatient Care Provider Order Entry: The Vanderbilt Experience 215
Randolph A. Miller, L. Russell Waitman, Sutin Chen, and S. Trent Rosenbloom
- CHAPTER 11 Decision Support for Patients 249
Holly B. Jimison, Paul P. Sher, Jennifer J.B. Jimison
- Index 263

Contributors

E. Andrew Balas, MD, PhD, FACMI

Dean and Professor, College of Health Sciences, Old Dominion University,
Norfolk, VA 23529, USA

Eta S. Berner, EdD, FACMI, FHIMSS

Professor, Health Informatics Program, Department of Health Services
Administration, School of Health Professions, University of Alabama at
Birmingham, Birmingham, AL 35294, USA

Suzanne Austin Boren, PhD MHA

Assistant Professor, Department of Health Management and Informatics,
School of Medicine, University of Missouri, Columbia, MO 65211, USA

Jerome H. Carter, MD

Chief Executive Officer, NT&M Informatics, Clinical Associate Professor
of Medical Education, Department of Medical Education, Morehouse
School of Medicine, Atlanta, GA 30324, USA

Sutin Chen, BS

Medical Student, Vanderbilt University, Nashville, TN 37231, USA

David C. Chhieng, MD, MBA, MSHI

Associate Professor, Department of Pathology, School of Medicine,
University of Alabama at Birmingham, Birmingham, AL 35294, USA

Paul Dexter, MD

Associate Professor, Clinical Medicine, Regenstrief Institute, and the
Department of Medicine, School of Medicine, Indiana University
Indianapolis, IN 46202, USA

R. Scott Evans, MS, PhD, FACMI

Senior Medical Informaticist, Professor Biomedical Informatics, LDS Hospital/Intermountain Healthcare, Salt Lake City, UT 84132, USA

Reed M. Gardner, PhD, FACMI

Professor, Department of Medical Informatics, Intermountain Healthcare, Department of Biomedical Informatics University of Utah, Salt Lake City, UT 84132, USA

Antoine Geissbuhler, MD, FACMI

Professor and Chairman, Division of Medical Informatics, Geneva University Hospitals, Geneva, Switzerland

Kenneth W. Goodman, PhD

Associate Professor of Medicine and Philosophy and Director, Bioethics Program, University of Miami, Miami, FL 33136, USA

J. Michael Hardin, PhD

Professor of Statistics and Associate Dean for Research, Culverhouse College of Commerce, University of Alabama, Tuscaloosa, AL 36497, USA

Peter J. Haug, MD, FACMI

Professor, Department of Biomedical Informatics, University of Utah, Senior Informaticist, Intermountain Healthcare, Salt Lake City, UT 84132, USA

Holly B. Jimison, PhD, FACMI

Associate Professor, Department of Medical Informatics and Clinical Epidemiology, Oregon Health and Science University, Portland, OR 97239, USA

Jennifer J.B. Jimison, MPA

Research Assistant, Department of Medical Informatics and Clinical Epidemiology, Oregon Health and Science University, Portland, OR 97239, USA

Tonya J. La Lande, MSHI

Systems Analyst, Bon Secours Richmond Health System, Richmond VA 23227, USA

Burke W. Mamlin, MD

Assistant Professor, Department of Clinical Medicine, Regenstrief Institute and Division of General Internal Medicine and Geriatrics, School of Medicine, Indiana University Indianapolis, IN 46202, USA

Clement J. McDonald, MD, FACP, FACMI

Director, Regenstrief Institute, Regenstrief Professor of Medical Informatics; Distinguished Professor, Department of Medicine, Indiana University School of Medicine, Indianapolis, IN 46202, USA

Randolph A. Miller, MD, FACP, FACMI

Donald A.B. and Mary M. Lindberg University Professor, Department of Biomedical Informatics, Vanderbilt University Medical Center, Vanderbilt University, Nashville, TN 37232, USA

J. Marc Overhage, MD, PhD, FACMI

Professor, Department of Medicine, School of Medicine, Indiana University and Regenstrief Institute, Indianapolis, IN 46202, USA

Beatriz H. Rocha, MD, PhD

Senior Medical Informaticist, Department of Medical Informatics, Intermountain Healthcare, Salt Lake City, UT 84132, USA

Roberto A. Rocha, MD, PhD

Assistant Professor, Department of Biomedical Informatics, University of Utah, School of Medicine, Salt Lake City, UT 84132, USA

S. Trent Rosenbloom, MD, MPH

Assistant Professor, Department of Biomedical Informatics, Assistant Professor, School of Nursing, Vanderbilt University Medical Center, Vanderbilt University, Nashville, TN 37232, USA

Paul P. Sher, MD

Consultant, Department of Medical Informatics and Clinical Epidemiology, Oregon Health and Science University, Portland, OR 97239, USA

S. Andrew Spooner, MD, FAAP

John Dustin Buckman Professor and Associate Professor of Pediatrics, Director, Division of General Pediatrics, Department of Pediatrics, University of Tennessee Health Science Center College of Medicine, Memphis, TN 38163; Director, Telehealth and Informatics, Le Bonheur Children's Medical Center, Memphis, TN 38103, USA

William Tierney, MD, FACMI

Research Scientist, Regenstrief Institute, Indianapolis, IN 46202,
Chancellor's Professor, Department of Medicine and Chief, Division of
General Internal Medicine and Geriatrics, Indiana University School of
Medicine, Indianapolis, IN 46202, USA

L. Russell Waitman, PhD

Assistant Professor, Department of Biomedical Informatics, Vanderbilt
University Medical Center, Vanderbilt University, Nashville, TN 37232,
USA

Section 1
Development and Evaluation
of Clinical Decision
Support Systems

1 Overview of Clinical Decision Support Systems*

ETA S. BERNER and TONYA J. LA LANDE

Introduction

Clinical decision support systems (CDSS) are computer systems designed to impact clinician decision making about individual patients at the point in time that these decisions are made. With the increased focus on the prevention of medical errors that has occurred since the publication of the landmark Institute of Medicine report, *To Err Is Human*, computer-based physician order entry (CPOE) systems, coupled with CDSS, have been proposed as a key element of systems' approaches to improving patient safety.¹⁻⁴ If used properly, CDSS have the potential to change the way medicine has been taught and practiced. This chapter will provide an overview of clinical decision support systems, summarize current data on the use and impact of clinical decision support systems in practice, and will provide guidelines for users to consider as these systems begin to be incorporated in commercial systems, and implemented outside the research and development settings. The other chapters in this book will explore these issues in more depth.

Types of Clinical Decision Support Systems

There are a variety of systems that can potentially support clinical decisions. Even Medline and similar healthcare literature databases can support clinical decisions. Decision support systems have been incorporated in healthcare information systems for a long time, but these systems usually have supported retrospective analyses of financial and administrative data.⁵ Recently, sophisticated data mining approaches have been proposed for similar retrospective analyses of both administrative and clinical data⁶ (see

* This chapter is an updated version of Chapter 36 in Ball MJ, Weaver C, Kiel J (eds). *Healthcare Information Management Systems*, Third Edition, New York: Springer-Verlag, 463-477, used with permission.

Chapter 3 for more details on data mining techniques). Although these retrospective approaches can be used to develop guidelines, critical pathways, or protocols to guide decision making at the point of care, such retrospective analyses are not usually considered to be CDSS. These distinctions are important because vendors often will advertise that their product includes decision support capabilities, but that may refer to the retrospective type of systems, not those designed to assist clinicians at the point of care. However, as the interest has increased in CDSS, more vendors have begun to incorporate these types of systems. Metzger and her colleagues^{7,8} have described CDSS using several dimensions. According to their framework, CDSS differ among themselves in the *timing* at which they provide support (before, during, or after the clinical decision is made) and how active or passive the support is, that is, whether the CDSS actively provides alerts or passively responds to physician input or patient-specific information. Finally, CDSS vary in how easy they are for busy clinicians to access.⁷ Although CDSS have been developed over the last thirty years, many of them have been stand-alone systems or part of noncommercial computer-based patient record systems. CDSS also differ in whether the information provided is general or specialty-based. In recent years, some of the originally noncommercial systems are now being more widely marketed, and other vendors are beginning to incorporate CDSS into their computer-based patient records and physician order entry systems.

Another categorization scheme for CDSS is whether they are knowledge-based systems, or nonknowledge-based systems that employ machine learning and other statistical pattern recognition approaches. Chapter 2 discusses the mathematical foundations of the knowledge-based systems, and Chapter 3 addresses the foundations of the statistical pattern recognition CDSS. In this overview, we will focus on the knowledge-based systems, and discuss some examples of other approaches, as well.

Knowledge-Based Clinical Decision Support Systems

Many of today's knowledge-based CDSS arose out of earlier expert systems research, where the aim was to build a computer program that could simulate human thinking.^{9,10} Medicine was considered a good domain in which these concepts could be applied. In the last twenty years, the developers of these systems have begun to adapt them so that they could be used more easily to support real-life patient care processes.¹¹ Many of the earliest systems were diagnostic decision support systems, which Miller and Geissbuhler discuss in Chapter 5. The intent of these CDSS was no longer to simulate an expert's decision making, but to assist the clinician in his or her own decision making. The system was expected to provide information for the user, rather than to come up with "the answer," as was the goal of earlier expert systems.¹² The user was expected to filter that information and to discard erroneous or useless information. The user was expected to

be active and to interact with the system, rather than just be a passive recipient of the output. This focus on the interaction of the user with the system is important in setting appropriate expectations for the way the system will be used.

There are three parts to most CDSS. These parts are the knowledge base, the inference or reasoning engine, and a mechanism to communicate with the user.¹³ As Spooner explains in Chapter 2, the knowledge base consists of compiled information that is often, but not always, in the form of if-then rules. An example of an if-then rule might be, for instance, IF a new order is placed for a particular blood test that tends to change very slowly, AND IF that blood test was ordered within the previous 48 hours, THEN alert the physician. In this case, the rule is designed to prevent duplicate test ordering. Other types of knowledge bases might include probabilistic associations of signs and symptoms with diagnoses, or known drug-drug or drug-food interactions.

The second part of the CDSS is called the inference engine or reasoning mechanism, which contains the formulas for combining the rules or associations in the knowledge base with actual patient data.

Finally, there has to be a communication mechanism, a way of getting the patient data into the system and getting the output of the system to the user who will make the actual decision. In some stand-alone systems, the patient data need to be entered directly by the user. In most of the CDSS incorporated into electronic medical records (EMR) systems, the data are already in electronic form and come from the computer-based patient record, where they were originally entered by the clinician, or may have come from laboratory, pharmacy, or other systems. Output to the clinician may come in the form of a recommendation or alert at the time of order entry, or, if the alert was triggered after the initial order was entered, systems of email and wireless notification have been employed.^{14,15}

CDSS have been developed to assist with a variety of decisions. The example above describes a system designed to provide support for laboratory test ordering. Diagnostic decision support systems have been developed to provide a suggested list of potential diagnoses to the users. The system might start with the patient's signs and symptoms, entered either by the clinician directly or imported from the EMR. The decision support system's knowledge base contains information about diseases and their signs and symptoms. The inference engine maps the patient signs and symptoms to those diseases and might suggest some diagnoses for the clinicians to consider. These systems generally do not generate only a single diagnosis, but usually generate a set of diagnoses based on the available information. Because the clinician often knows more about the patient than can be put into the computer, the clinician will be able to eliminate some of the choices. Most of the diagnostic systems have been stand-alone systems, but the Wizorder system, developed at Vanderbilt University, has a diagnostic system that runs in the background, taking its information from the data

already in the EMR.¹⁶ This system has been incorporated into the McKesson Horizon Clinicals™ system. The use of CDSS at Vanderbilt is described in detail by Miller and his colleagues in Chapter 10.

Other systems can provide support for medication orders, a major cause of medical errors.^{1,17} The input for the system might be the patient's laboratory test results for the blood level of a prescribed medication. The knowledge base might contain values for therapeutic and toxic blood concentrations of the medication and rules on what to do when a toxic level of the medication is reached. If the medication level was too high, the output might be an alert to the physician.¹⁷ There are CDSS that are part of computerized physician order entry (CPOE) systems that take a new medication order and the patient's current medications as input, the knowledge base might include a drug database and the output would be an alert about drug interactions so that the physician could change the order. Similarly, input might be a physician's therapy plan, where the knowledge base would contain local protocols or nationally accepted treatment guidelines, and the output might be a critique of the plan compared to the guidelines.¹⁸ Some hospitals that have implemented these systems allow the user to override the critique or suggestions, but often the users are required to justify why they are overriding it. The structure of the CDSS knowledge base will differ depending on the source of the data and the uses to which they are put. The design considerations, especially vocabulary issues, are not trivial. The challenges of CDSS design are discussed in more detail in Chapter 4.

Nonknowledge-Based Clinical Decision Support Systems

Unlike knowledge-based decision support systems, some of the nonknowledge-based CDSS use a form of artificial intelligence called machine learning, which allows the computer to learn from past experiences and/or to recognize patterns in the clinical data.¹⁹ Artificial neural networks and genetic algorithms are two types of nonknowledge-based systems.¹⁹

Artificial Neural Networks

Research in neural networks has been going on since the 1940s.²⁰ Artificial neural networks (ANN) simulate human thinking and learn from examples.¹⁹ An ANN consists of nodes called neurodes (which correspond to neurons) and weighted connections (which correspond to nerve synapses) that transmit signals between the neurodes in a unidirectional manner.^{19,21} An ANN contains 3 layers, which include the input layer, output layer, and hidden layer.¹⁹ The input layer is the data receiver and the output layer communicates the results, while the hidden layer processes the incoming data and determines the results.¹⁹

This structure bears some similarities to the knowledge-based decision support systems, but rather than having a knowledge base derived from the

medical literature or from an expert clinician's knowledge, the ANN analyzes the patterns in the patient data, to derive the associations between the patient's signs and symptoms and a diagnosis. Many of the knowledge-based CDSS cover a wide range of diseases. For instance, the input may be the signs and symptoms exhibited by a patient and the output may be the possible diseases the patient may have. Neural networks often focus on a more narrow range of signs and symptoms, for instance, those associated with a single disease, such as myocardial infarction.²²

These systems can learn from examples when supplied with known results for a large amount of data.²¹ The system will study this information, make guesses for the correct output, compare the guesses to the given results, find patterns that match the input to the correct output, and adjust the weights of the connections between the neurodes accordingly, in order, to produce the correct results.²¹ This iterative process is known as training the artificial network.²¹ In the example with myocardial infarction, for instance, the data including a variety of signs and symptoms from large numbers of patients who are known to either have or not have a myocardial infarction can be used to train the neural network. Once the network is trained, i.e., once the weighted associations of signs and symptoms with the diagnosis are determined, the system can be used on new cases to determine if the patient has a myocardial infarction.

There are many advantages and disadvantages to using artificial neural networks. Advantages include eliminating the need to program IF-THEN rules and eliminating the need for direct input from experts.¹⁹ ANNs can also process incomplete data by inferring what the data should be and can improve every time they are used because of their dynamic nature.¹⁹ ANNs also do not require a large database to make predictions about outcomes, but the more comprehensive the training data set is, the more accurate the ANN is likely to be.²¹ Even though all of these advantages exist, there are some disadvantages. The training process involved can be time consuming.¹⁹ ANNs follow a statistical pattern recognition approach to derive their formulas for weighting and combining data. The resulting formulas and weights are often not easily interpretable, and the system cannot explain or justify why it uses certain data the way it does, which can make the reliability and accountability of these systems a concern.¹⁹

Despite the above concerns, artificial neural networks have many applications in the medical field. In a review article on the use of neural networks in health care, Baxt provides a chart that shows various applications of ANNs, which include the diagnosis of appendicitis, back pain, dementia, myocardial infarction, psychiatric emergencies, sexually transmitted diseases, skin disorders, and temporal arteritis.²³ Study results have shown that ANNs' diagnostic predictions for pulmonary embolisms were as good as, or better than, physicians' predictions.²⁴ Another study also showed that neural networks did a better job than two experienced cardiologists in detecting acute myocardial infarction in electrocardiograms with concomitant left

bundle branch block.²⁵ Studies have also shown that ANNs can predict which patients are at high risk for cancers such as oral cancer.²⁶ The studies described in Baxt's chart illustrate other applications of ANNs, including predicting outcomes for things such as surgery recovery, liver transplants, cardiopulmonary resuscitation, and heart valve surgery, as well as the analysis of waveforms of electrocardiograms (ECGs) and electroencephalograms (EEGs).²³

Genetic Algorithms

Another nonknowledge-based method used to create CDSS is a genetic algorithm (GA). GAs were developed in the 1940s by John Holland at the Massachusetts Institute of Technology, and are based on the evolutionary theories by Darwin that dealt with natural selection and survival of the fittest.¹⁹ Just as species change to adapt to their environment, "GAs 'reproduce' themselves in various recombinations in an effort to find a new recombinant that is better adapted than its predecessors. (page 239)."¹⁹ In other words, without any domain-specific knowledge, components of random sets of solutions to a problem are evaluated, the best ones are kept and are then recombined and mutated to form the next set of possible solutions to be evaluated, and this continues until the proper solution is discovered.²⁷ The fitness function is used to determine which solutions are good and which ones should be eliminated.¹⁹ GAs are similar to neural networks in that they derive their knowledge from patient data.

Genetic algorithms have also been applied in health care, but there are fewer examples of this type of CDSS than those based on neural networks. However, GAs have proved to be a helpful aid in the diagnosis of female urinary incontinence.²⁸

Although, as Hardin and Chhieng describe in Chapter 3, research has shown that CDSS based on pattern recognition and machine learning approaches may be more accurate than the average clinician in diagnosing the targeted diseases, many physicians are hesitant to use these CDSS in their practice because the reasoning behind them is not transparent.²⁴ Most of the systems that are available today involve knowledge-based systems with rules, guidelines, or other compiled knowledge derived from the medical literature. The research on the effectiveness of CDSS has come largely from a few institutions where these systems were developed.

Effectiveness of Clinical Decision Support Systems

Clinical decision support systems have been shown to improve both patient outcomes, as well as the cost of care. Because many of the published studies have come out of a limited number of institutions including LDS Hospital,

Regenstrief Medical Institute and, more recently, Vanderbilt University. Chapter 8 describes the CDSS deployed in the HELP system at LDS Hospital, Chapter 9 describes the system at the Regenstrief Institute, and Chapter 10 describes Vanderbilt's system. In addition, there are an increasing number of studies from other places, that have shown positive impact.^{17,29-33} The systems can minimize errors by alerting the physician to potentially dangerous drug interactions, and the diagnostic programs have also been shown to improve physician diagnoses.³³⁻³⁶ The reminder and alerting programs can potentially minimize problem severity and prevent complications. They can warn of early adverse drug events that have an impact on both cost and quality of care.^{4,29,37,38,39} These data have prompted the Leapfrog Group and others to advocate their use in promoting patient safety.³ As described in the chapters in Section 2 of this book, most of the studies that have shown the strongest impact on reducing medication errors have been done at institutions with very sophisticated, internally developed systems, and where use of an EMR, CPOE, and CDSS are a routine and accepted part of the work environment. As more places that do not have that cultural milieu, or a good understanding of the strengths and limitations of the systems, begin to adopt CDSS, integration of these systems may prove more difficult.⁴⁰

Several published reviews of CDSS have emphasized the dearth of evidence of similar effectiveness on a broader scale and have called for more research, especially qualitative research, that elucidates the factors which lead to success outside the development environment.^{41,42} Studies of the Leeds University abdominal pain system, an early CDSS for diagnosis of the acute abdomen, showed success in the original environment and much more limited success when the system was implemented more broadly.^{43,44} As Chapter 7 shows, while the evidence is increasing, there are still limited systematic, broad-scale studies of the effectiveness of CDSS. Not only is there a lack of studies on the impact of the diffusion of successful systems, but also there are still few places utilizing the systems themselves.^{45,46} The KLAS research and consulting firm conducted an extensive survey of the sites that had implemented CPOE systems.⁴⁶ As KLAS defines these systems, CPOE systems usually include CDSS that were defined as, "... alerting, decision logic and knowledge tools to help eliminate errors during the ordering process."⁴⁶ Although most of the CPOE systems provide for complex decision support, the results of the KLAS survey showed that most sites did not use more than 10 alerts and that many sites did not use any of the alerting mechanisms at order entry. These data on system use outside the research and development sites underscore the fact that despite evidence that CDSS have many benefits, as the KLAS report states, "The use of active complex alerts is in its infancy."⁴⁶

Metzger and McDonald report anecdotal case studies of successful implementation of CDSS in ambulatory practices.⁸ While such descriptions can motivate others to adopt CDSS, they are not a substitute for

systematic evaluation of implementation in a wide range of settings. Unfortunately, when such evaluations are done, the results have sometimes been disappointing. A study incorporating guideline-based decision support systems in 31 general practice settings in England found that, although care was not optimal before implementing the computer-based guidelines, there was little change in health outcomes after the system was implemented. Further examination showed that, although the guideline was triggered appropriately, clinicians did not go past the first page and essentially did not use it.¹⁸ Another study found that clinicians did not follow the guideline advice because they did not agree with it.⁴⁷

There is a body of research that has shown that physicians have many unanswered questions during the typical clinical encounter.^{48,49} This should provide an optimal opportunity for the use of CDSS, yet a study tracking the use of a diagnostic system by medical residents indicated very little use.⁵⁰ This is unusual given that this group of physicians in training should have even more “unanswered questions” than more experienced practitioners, but this may be partially explained by the fact that the system was a stand-alone system not directly integrated into the workflow. Also, Teich et al. suggest that reminder systems and alerts usually work, but systems that challenge the physicians’ judgment, or require them to change their care plans, are much more difficult to implement.⁵¹ A case study of a CDSS for notification of adverse drug events supports this contention. The study showed that despite warnings of a dangerous drug level, the clinician in charge repeatedly ignored the advice. The article describes a mechanism of alerting a variety of clinicians, not just the patient’s primary physician, to assure that the alerts receive proper attention.¹⁷ Bria made analogies to making some alerts impossible to ignore. He used the example of the shaking stick in an airplane to alert the pilots to really serious problems.⁵² In addition to the individual studies, Kawamoto et al.⁵³ examined factors associated with CDSS success across a variety of studies. They found that four factors were the main correlates of successful CDSS implementation. The factors were:

1. providing alerts/reminders automatically as part of the workflow;
2. providing the suggestions at a time and location where the decisions were being made;
3. providing actionable recommendations; and
4. computerizing the entire process.

Thus, although these systems can potentially influence the process of care, if they are not used, they obviously cannot have an impact. Integration into both the culture and the process of care is going to be necessary for these systems to be optimally used. Institutions that have developed such a culture provide a glimpse of what is potentially possible (see Chapters 8–10). However, Wong et al., in an article published in 2000, suggest that the incentives for use are not yet aligned to promote wide-scale adoption

of CDSS.⁴⁰ Those incentives may be changing, but these systems are not equally attractive to all stakeholders. As Doolan and Bates³⁸ illustrate, hospital administrators are beginning to see advantages to adoption of CDSS and other clinical computing applications, but at this point in time the perceived disadvantages for physicians may loom larger than the advantages. There are several reasons why implementation of CDSS is challenging.

Some of the problems include issues of how the data are entered. Other issues include the development and maintenance of the knowledge base and issues around the vocabulary and user interface. Finally, since these systems may represent a change in the usual way patient care is conducted, there is a question of what will motivate their use, which also relates to how the systems are evaluated.

Implementation Challenges

The first issue concerns data entry, or how the data will actually get into the system. Some systems require the user to query the systems and/or enter patient data manually. Not only is this “double data entry” disruptive to the patient care process, it is also time consuming, and, especially in the ambulatory setting, time is scarce. It is even more time consuming if the system is not mobile and/or requires a lengthy logon. Much of this disruption can be mitigated by integrating the CDSS with the hospital information system and EMR. As mentioned above, several commercial products have integrated decision support capabilities. What that means is if the data are already entered into the medical record, the data are there for the decision support system to act upon, and, in fact, many systems are potentially capable of drawing from multiple ancillary systems as well. This is a strength, but not all clinical decision support systems are integrated, and without technical standards assuring integration of ancillary systems, such linkages may be difficult. There are also a number of stand-alone systems, some of the diagnostic systems and some drug interaction systems, for example. This means that patient data have to be entered twice—once into the medical record system, and again, into the decision support system. For many physicians, this double data entry can limit the usefulness of such systems.

A related question is who should enter the data in a stand-alone system or even in the integrated hospital systems. Physicians are usually the key decision makers, but they are not always the person who interacts with the hospital systems. One of the reasons for linking CDSS with physician order entry is that it is much more efficient for the physician to receive the alerts and reminders from decision support systems. The issue concerns not just order entry, but also mechanisms of notification. The case study mentioned earlier described a situation where the physician who received the alert ignored it.¹⁷ These systems can be useful, but their full benefits cannot be

gained without collaboration between the information technology professionals and the clinicians.

Although it might not seem that vocabularies should be such a difficult issue, it is often only when clinicians actually try to use a system, either a decision support system or computer-based patient record or some other system with a controlled vocabulary, that they realize either the system cannot understand what they are trying to say or, worse yet, that it uses the same words for totally different concepts or different words for the same concept. The problem is there are no standards that are universally agreed upon for clinical vocabulary and, since most of the decision support systems have a controlled vocabulary, errors can have a major impact (see Chapter 4 for more discussion on vocabulary and other design issues).

Future Uses of Clinical Decision Support Systems

Despite the challenges in integrating CDSS, when properly used they have the potential to make significant improvements in the quality of patient care. While more research still needs to be done evaluating the impact of CDSS outside the development settings and the factors that promote or impede integration, it is likely that increased commercialization will continue. CDSS for non-clinician users such as patients are likely to grow as well (see Chapter 11). There is increasing interest in clinical computing and, as handheld and mobile computing become more widely adopted, better integration into the process of care may be easier⁵⁴ In addition, the concerns over medical errors and patient safety are prompting a variety of initiatives that will lead to increased incorporation of CDSS. Physicians are legally obligated to practice in accordance with the standard of care, which at this time does not mandate the use of CDSS. However, that may be changing. The issue of the use of information technology in general, and clinical decision support systems in particular, to improve patient safety, has received a great deal of attention recently.^{1,2,55} Healthcare administrators, payers, and patients, are concerned, now more than ever before, that clinicians use the available technology to reduce medical errors. The Leapfrog Group³ has advocated physician order entry (with an implicit coupling of CDSS to provide alerts to reduce medication errors) as one of their main quality criteria.

Even if the standard of care does not yet require the use of such systems, there are some legal and ethical issues that have not yet been well addressed (see Chapter 6 for a fuller discussion of these issues). One interesting legal case that has been mentioned in relation to the use of technology in health care is the Hooper decision. This case involved two tugboats (the T.J. Hooper and its sister ship) that were pulling barges in the 1930s when radios (receiving sets) were available, but not widely used on tugboats. Because the boats did not have a radio, they missed storm warnings and their cargo

sank. The barge owners sued the tugboat company, even though the tugboat captains were highly skilled and did the best they could under the circumstances to salvage their cargo. They were found liable for not having the radio, even though it was still not routinely used in boats. Parts of the following excerpt from the Hooper decision have been cited in other discussions of CDSS⁵⁶

... whole calling may have unduly lagged in the adoption of new and available devices. It never may set its own tests, however persuasive be its usages. Courts must in the end say what is required; there are precautions so imperative that even their universal disregard will not excuse their omission. But here there was no custom at all as to receiving sets; some had them, some did not; the most that can be urged is that they had not yet become general. Certainly in such a case we need not pause; when some have thought a device necessary, at least we may say that they were right, and the others too slack.⁵⁷

It has been suggested that as CDSS and other advanced computer systems become more available, the Hooper case may not only provide legal precedent for liability for failure to use available technology, but the legal standard of care may also change to include using available CDSS.⁵⁸ Since this area is still new, it is not clear what type of legal precedents will be invoked for hospitals that choose to adopt, or avoid adopting, CDSS. One legal scholar suggests that while the use of CDSS may lower a hospital's risk of medical errors, healthcare systems may incur new risks if the systems either cause harm or are not implemented properly.⁵⁹ In any case, there are some guidelines that users can follow that may help ensure more appropriate use of CDSS.

Guidelines for Selecting and Implementing Clinical Decision Support Systems†

Osheroff et al. offer practical suggestions for steps to be taken in the implementation of CDSS.⁶⁰ The guidelines below address other issues such as those involved in selecting CDSS, interacting with vendors, and assuring that user expectations for CDSS are appropriate. They also include legal and ethical issues that are discussed in more detail in Chapter 6.

Assuring That Users Understand the Limitations

In 1986, Brannigan and Dayhoff highlighted the often different philosophies of physicians and software developers.⁶¹ Brannigan and Dayhoff mention that physicians and software developers differ in regard to how

† Significant parts of this section and smaller parts of other sections were reprinted with permission from Berner ES. Ethical and Legal Issues in the Use of Clinical Decision Support Systems. *J. Healthcare Information Management*, 2002;16(4):34–37.

“perfect” they expect their “product” to be when it is released to the public.⁶¹ Physicians expect perfection from themselves and those around them. Physicians undergo rigorous training, have to pass multiple licensing examinations, and are held in high esteem by society for their knowledge and skills. In contrast, software developers often assume that initial products will be “buggy” and that eventually most errors will be fixed, often as a result of user feedback and error reports. There is usually a version 1.01 of almost any system almost as soon as version 1.0 has reached most users. Because a CDSS is software that in some ways functions like a clinician consultant, these differing expectations can present problems, especially when the knowledge base and/or reasoning mechanism of the CDSS is not transparent to the user. The vendors of these systems have an obligation to learn from the developers, and to inform the clinicians using the CDSS of its strengths and limitations.

Assuring That the Knowledge Is From Reputable Sources

Users of CDSS need to know the source of the knowledge if they purchase a knowledge-based system. What rules are actually included in the system and what is the evidence behind the rules? How was the system tested before implementation? This validation process should extend not just to testing whether the rules fire appropriately in the face of specific patient data (a programming issue), but also to whether the rules themselves are appropriate (a knowledge-engineering issue). Sim et al. advocate the use of CDSS to promote evidence-based medical practice, but this can only occur if the knowledge base contains high quality information.⁶²

Assuring That the System Is Appropriate for the Local Site

Vendors need to alert the client about idiosyncrasies that are either built into the system or need to be added by the user. Does the clinical vocabulary in the system match that in the EMR? What are the normal values assumed by a system alerting to abnormal laboratory tests, and do they match those at the client site? In fact, does the client have to define the normal values as well as the thresholds for the alerts?

The answers to the questions about what exactly the user is getting are not always easy to obtain. A few years ago, the chapter authors conducted a survey of the nine vendors listed in the top 100 companies of the annual vendor revenue survey of *Healthcare Informatics*. These companies also indicated that their EMR systems contained decision support functionality.^{63,64} We began by reviewing the vendor Web sites to see how much information about the CDSS they contained. If we could not find answers to our

questions on the Web site, we telephoned the vendors to find an appropriate person to answer our questions.

The survey asked vendors whether they provided a knowledge base/medical logic model to their customers. If they answered yes, they were asked what the knowledge source was, if the knowledge base was updated, how often the knowledge base was updated, and if there was an additional charge for the updates. If they answered no to providing a knowledge base, they were asked if they provided templates for the user to develop rules, if there was an additional charge for these templates, how much effort was involved for the customer to build these rules, and whether they provided mechanisms to obtain/buy rules from somewhere else, and if there was a charge.

None of the vendor Web sites contained answers to all of the questions on the survey. The knowledge source was given on one of the nine vendor Web sites, and two of the nine vendor Web sites indicated that they provided templates to develop the rules. All nine of the vendors needed to be contacted to obtain additional information. Obtaining this information turned out to be a more challenging task than expected.

Three of the vendor representatives with whom we spoke were very helpful and open to answering the questions. The other six did not know the answers to some or all of the questions and said they would refer our questions to someone else who would call us back. After waiting a month with no response from the vendors, we utilized our personal contacts with users of five of the remaining systems to request either answers or a referral to another vendor contact. Two of those contacts returned answers to most of our questions, leaving us with four companies for whom we could not obtain answers to most of our questions.

The results of our survey are based on the full answers to the questionnaire from four of the nine clinical decision support vendors, as well as the information that was obtained from the Web sites and the partial answers from one of the five remaining vendors. The results show that five of the nine vendors provide a knowledge base/medical logic model. Two of the five vendors said their knowledge base comes from rules they developed based on experience with their customer base in the past. One uses a physician order entry system and knowledge source that was developed and is currently used by a well-known academic medical center, and one of the five vendors did not know the source of their knowledge base. Three of the five vendors said they update their knowledge base, one does not perform updates, and two vendor representatives did not know if their knowledge base was regularly updated. Two of the vendors who said they provided updates were not sure how often they occurred.

The results also show that seven of the nine vendors provide templates to develop the rules. Three of these seven vendors did not have an answer to the question about the amount of effort that is involved for the customer in building these rules. Four of the seven vendors said it did not take much

effort to build the rules with the templates. In fact, one vendor pointed out that the time consuming part was researching, validating, and approving new rules. Also, one vendor said they provide a community Web site for their customers to post and share rules. Four vendors said they did not provide mechanisms to obtain/buy rules from somewhere else, one vendor said clients could obtain rules from professional committees and organizations, and four vendors did not have an answer to this question.

When users ask questions like those in our survey, they may find, as we did, that the decision support system provided is really just an expert system shell and that local clinicians need to provide the “knowledge” that determines the rules. For some systems, an effort has been made to use standards that can be shared among different sites, for example, the Arden syntax for medical logic modules,⁶⁵ but local clinicians must still review the logic in shared rules to assure that they are appropriate for the local situation. Using in-house clinicians to determine the rules in the CDSS can assure its applicability to the local environment, but that means extensive development and testing must be done locally to assure the CDSS operates appropriately. Often a considerable amount of physician time is needed. Without adequate involvement by clinicians, there is a risk that the CDSS may include rules that are inappropriate for the local situation, or, if there are no built-in rules, that the CDSS may have only limited functionality. On the other hand, local development of the logic behind the rules may also mean that caution should be exercised if the rules are used at different sites. The important thing is for the user to learn at the outset what roles the vendor and the client will have to play in the development and maintenance of the systems. Based on our experience, and despite the fact that many of the vendors did make an effort to provide the answers for us, there were still many important questions for which we could not easily obtain the information. These results can help to explain the findings from the KLAS survey of CPOE users, which involved users of systems of many of the same vendors we surveyed. Although these systems have decision support capabilities, the effort involved in customizing the CDSS for the local site may be considerable, and the result may be that CDSS capabilities are underutilized.⁴⁶

Assuring That Users Are Properly Trained

Just as the vendor should inform the client how much work is needed to get the CDSS operational, the vendor should also inform the client how much technical support and/or clinician training is needed for physicians to use the system appropriately and/or understand the systems’ recommendations. It is not known whether the users of some CDSS need special clinical expertise to be able to use it properly, in addition to the mechanics of training on the use of the CDSS. For instance, systems that base their

recommendations on what the user enters directly or on what was entered into the medical record by clinicians have been shown to reach faulty conclusions or make inappropriate recommendations if the data on which the CDSS bases its recommendations are incomplete or inaccurate.⁶⁶ Also, part of the reason for integrating CDSS with physician order entry is that it is assumed the physician has the expertise to understand, react to, and determine whether to override the CDSS recommendation. Diagnostic systems, for instance, may make an appropriate diagnostic suggestion that the user fails to recognize.^{36,67,68} Thus, vendors of CDSS need to be clear about what expertise is assumed in using the system, and those who implement the systems need to assure that only the appropriate users are allowed to respond to the CDSS advice.

As these systems mature and are more regularly integrated into the healthcare environment, another possible concern about user expertise arises. Will users lose their ability to determine when it is appropriate to override the CDSS? This “de-skilling” concern is similar to that reported when calculators became commonplace in elementary and secondary education, and children who made errors in using the calculator could not tell that the answers were obviously wrong. The solution to the problem is not to remove the technology, but to remain alert to both the positive and negative potential impact on clinician decision making.

Monitoring Proper Utilization of the Installed Clinical Decision Support Systems

Simply having a CDSS installed and working does not guarantee that it will be used. Systems that are available for users if they need them, such as online guidelines or protocols, may not be used if the user has to choose to consult the system, and especially if the user has to enter additional data into the system. Automated alerting or reminder systems that prompt the user can address the issue of the user not recognizing the need for the system, but another set of problems arises with the more automated systems. They must be calibrated to alert the user often enough to prevent serious errors, but not so frequently that they will be ignored eventually. As mentioned earlier, there have been reports of CDSS triggering an alert that the patient’s physician ignored.¹⁷ What this means is that testing the system with the users, and monitoring its use, is essential for the CDSS to operate effectively in practice as well as in theory.

Assuring the Knowledge Base Is Monitored and Maintained

Once the CDSS is operational at the client site, a very important issue involves the responsibility for updating the knowledge base in a timely manner. New diseases are discovered, new medications come on the

market, and issues like the threat of bioterrorist actions prompt a need for new information to be added to the CDSS. Does the vendor have an obligation to provide regular knowledge updates? Such maintenance can be an expensive proposition given both rapidly changing knowledge and systems with complex rule sets. Who is at fault if the end user makes a decision based on outdated knowledge, or, conversely, if updating one set of rules inadvertently affects others, causing them to function improperly? Such questions were raised over 20 years ago,⁶⁹ but because CDSS are still not in widespread use, the legal issues have not really been tested or clarified.

The Food and Drug Administration (FDA) is charged with device regulation and has recently begun to reevaluate its previous policy on software regulation. Up to now, many stand-alone CDSS have been exempt from FDA device regulation because they required “competent human intervention” between the CDSS’ advice and anything being done to the patient.⁷⁰ Even if the rules change and CDSS are required to pass a pre-market approval process, monitoring would need to be ongoing to ensure the knowledge does not get out of date, and that what functioned well in the development process still functions properly at the client site. For this reason, local software review committees, which would have the responsibility to monitor local software installations for problems, obsolete knowledge, and harm as a result of use, have been advocated.⁷¹

Conclusion

There is now growing interest in the use of CDSS.⁷² More vendors of information systems are incorporating them. As skepticism about the usefulness of computers for clinical practice decreases, the wariness about accepting the CDSS’ advice, that many clinicians currently exhibit, is likely to decrease. As research has shown, if CDSS are available and convenient, and if they provide what appears to be good information, they are likely to be heeded by clinicians. The remaining chapters in this book explore the issues raised here in more depth. Underlying all of them is the perspective that, as CDSS become widespread, we must continue to remember that the role of the computer should be to enhance and support the human who is ultimately responsible for the clinical decisions.

References

1. Kohn LT, Corrigan JM, Donaldson MS, eds. *To err is human: building a safer health system*. Washington, DC: National Academy Press; 2000.
2. Institute of Medicine. *Crossing the quality chasm: a new health system for the 21st century*. Washington, DC: National Academy Press; 2001.
3. The Leapfrog Group. www.leapfroggroup.org

4. Bates DW, Leape LL, Cullen DJ, et al. Effect of computerized physician order entry and a team intervention on prevention of serious medical errors. *JAMA* 1998;280:1311–1316.
5. Oliveira, J. A shotgun wedding: business decision support meets clinical decision support. *J Healthc Inf Manage* 2002;16:28–33.
6. DeGruy KB. Healthcare applications of knowledge discovery in databases. *J. Healthc Inf Manage* 2000;14:59–69.
7. Perreault LE, Metzger JB. A pragmatic framework for understanding clinical decision support. In: Middleton, B, ed. *Clinical Decision Support Systems*. *J Healthc Inf Manage* 1999;13:5–21.
8. Metzger J, MacDonald K. *Clinical decision support for the independent physician practice*. Oakland: California Healthcare Foundation; 2002.
9. Shortliffe EH, Axline SG, Buchanan BG, Merigan TC, Cohen SN. An artificial intelligence program to advise physicians regarding antimicrobial therapy. *Comput Biomed Res* 1973;6(6):544–560.
10. Miller RA, Pople HE Jr., Myers JD. Internist-I, an experimental computer-based diagnostic consultant for general internal medicine. *N Engl J Med* 1982; 307:468–476.
11. Miller RA, McNeil MA, Challinor S, Masarie FE Jr, Myers JD. Internist-I, quick medical reference project-status report. *West J Med* 1986;145:816–822.
12. Miller RA, Masarie FE Jr. The demise of the “Greek Oracle” model for medical diagnostic systems. *Methods Inf Med* 1990;29:1–2.
13. Tan JKH, Sheps S. *Health decision support systems*. Gaithersburg: Aspen Publishers, Inc.; 1998.
14. Kuperman GJ, Teich JM, Bates DW, et al. Detecting alerts, notifying the physician, and offering action items: a comprehensive alerting system. *Proc AMIA Annu Fall Symp* 1996:704–708.
15. Shabot MM, LoBue M, Chen J. Wireless clinical alerts for physiologic, laboratory and medication data. *Proc AMIA Symp* 2000:789–793.
16. Geissbuhler A, Miller RA. Clinical application of the UMLS in a computerized order entry and decision-support system. *Proc AMIA Symp* 1998:320–324.
17. Galanter WL, DiDomenico RJ, Polikaitis A. Preventing exacerbation of an ADE with automated decision support. *J Healthc Inf Manage* 2002;16(4):44–49.
18. Eccles M, McColl E, Steen N, et al. Effect of computerised evidence based guidelines on management of asthma and angina in adults in primary care: cluster randomised controlled trial. *BMJ* 2002;325:941.
19. Marakas GM. *Decision support systems in the 21st century*. Upper Saddle River: Prentice Hall; 1999.
20. Devillers J, ed. *Neural networks in QSAR and drug design*. London: Academic Press; 1996.
21. Hirshberg A, Adar R. Artificial neural networks in medicine. *Isr J Med Sci* 1997;33:700–702.
22. Cross S, Harrison R, Kennedy RL. Introduction to neural networks. *Lancet* 1995;346:1075–1079.
23. Baxt WG. Application of artificial neural networks to clinical medicine. *Lancet* 1995;346:1135–1138.
24. Holst H, Astrom K, Jarund A, et al. Automated interpretation of ventilation-perfusion lung scintigrams for the diagnosis of pulmonary embolism using artificial neural networks. *Eur J Nucl Med* 2000;27:400–406.

25. Olsson SE, Ohlsson M, Ohlin H, Edenbrandt L. Neural networks—a diagnostic tool in acute myocardial infarction with concomitant left bundle branch block. *Clin Physiol Funct Imaging* 2002;22:295–299.
26. Naguib RNG, Sherbet GV, eds. *Artificial neural networks in cancer diagnosis, prognosis, and patient management*. Boca Raton: CRC Press; 2001.
27. Levin M. Use of genetic algorithms to solve biomedical problems. *MD Comput* 1995;12:193–199.
28. Laurikkala J, Juhola M, Lammi S, Viikki K. Comparison of genetic algorithms and other classifications methods in the diagnosis of female urinary incontinence. *Methods Inf Med* 1999;38:125–131.
29. Doolan DF, Bates DW, James BC. The use of computers for clinical care: a case series of advanced U.S. sites. *J Am Med Inform Assoc* 2003;10:94–107.
30. Berner ES, Houston TK, Ray MN, et al. Improving ambulatory prescribing safety with a handheld decision support system: a randomized controlled trial. *J Am Med Inform Assoc* 2006;13:171–179.
31. Thomas NJ. ISABEL. *Crit Care* 2002;7:99–100.
32. Ramnarayan P, Tomlinson A, Rao A, Coren M, Winrow A, Britto J. ISABEL: a web-based differential diagnostic aid for paediatrics: results from an initial performance evaluation. *Arch Dis Child* 2003;88:408–413.
33. Ramnarayan P, Kapoor RR, Coren M, et al. Measuring the impact of diagnostic decision support on the quality of decision-making: development of a reliable and valid composite score. *J Am Med Inform Assoc* 2003;10:563–572.
34. Berner ES, Maisiak RS, Cobbs CG, Taunton OD. Effects of a decision support system on physicians' diagnostic performance. *J Am Med Inform Assoc* 1999;6:420–427.
35. Berner ES, Maisiak RS. Influence of case and physician characteristics on perceptions of decision support systems. *J Am Med Inform Assoc* 1999;6:428–434.
36. Friedman CP, Elstein AS, Wolf FM, et al. Enhancement of clinicians' diagnostic reasoning by computer-based consultation. A multisite study of 2 systems. *JAMA* 1999;282:1851–1856.
37. Evans RS, Pestotnik SL, Classen DC, et al. A computer-assisted management program for antibiotics and other antiinfective agents. *N Engl J Med* 1998;338:232–238.
38. Doolan DF, Bates DW. Computerized physician order entry systems in hospitals: mandates and incentives. *Health Aff* 2002;21:180–188.
39. Kucher N, Koo S, Quiroz R, et al. Electronic alerts to prevent venous thromboembolism among hospitalized patients. *N Engl J Med* 2005;352:969–977.
40. Wong HJ, Legnini MW, Whitmore HH. The diffusion of decision support systems in healthcare: are we there yet? *J Healthc Manage* 2000;45:240–249; discussion 249–253.
41. Garg AX, Adhikari NK, McDonald H, et al. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *JAMA* 2005;293:1223–1238.
42. Kaplan B. Evaluating informatics applications—clinical decision support systems literature review. *Int J Med Inform* 2001;64:15–37.
43. de Dombal FT. The diagnosis of acute abdominal pain with computer assistance: worldwide perspective. *Ann Chir* 1991;45:273–277.
44. Adams ID, Chan M, Clifford PC, et al. Computer aided diagnosis of acute abdominal pain: a multicentre study. *BMJ* 1986;293:800–804.

45. Burt CW, Hing E. Use of computerized clinical support systems in medical settings. United States, 2001–2003. Advance data from vital and health statistics; no. 353. Hyattsville, MD: National Center for Health Statistics; 2005.
46. KLAS Research and Consulting Firm. CPOE digest (computerized physician order entry). KLAS Enterprises, LLC; 2003.
47. Keefe B, Subramanian U, Tierney WM, et al. Provider response to computer-based care suggestions for chronic heart failure. *Med Care* 2005;43:461–465.
48. Covell DG, Uman GC, Manning PR. Information needs in office practice: Are they being met? *Ann Intern Med* 1985;103:596–569.
49. Gorman PN, Helfand M. Information seeking in primary care: How physicians choose which clinical questions to pursue and which to leave unanswered. *Med Decis Making* 1995;15:113–119.
50. Berner ES, Maisiak RS, Phelan ST, Kennedy JI, Heudebert GR, Young KR. Use of a diagnostic decision support system by internal medicine residents. Unpublished study; 2002.
51. Teich JM, Merchia PR, Schmiz JL, Kuperman GJ, Spurr CD, Bates DW. Effects of computerized physician order entry on prescribing practices. *Arch Intern Med*. 2000;160:2741–2747.
52. Bria WF. Clinical decision support and the parable of the stick shaker. *J Healthc Info Manage* 2005;19:8–9.
53. Kawamoto K, Houlihan CA, Bala EA, Lobach DF. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. *BMJ* 2005;330:765.
54. Turisco F, Case J. Wireless and mobile computing. Oakland, CA: California Healthcare Foundation; 2001.
55. Morris AH. Decision support and safety of clinical environments. *Qual Saf Health Care* 2002;11:69–75.
56. Weed LL, Weed L. Opening the black box of clinical judgment—an overview. *BMJ* 1999;319:1279. Data Supplement—Complete Version. Available from: <http://www.bmj.org/cgi/content/full/319/7220/1279/DC1>.
57. The T.J. Hooper. 60 F.2d 737 (2d Cir. 1932).
58. Osler, Harkins, Harcourt Law Firm. Ten commandments of computerization. Reprinted with permission by the Canadian Information Processing Society; 1992. Available from: <http://www.cips.ca/it/resources/default.asp?load=practices>.
59. Terry NP. When the machine that goes “ping” causes harm: default torts rules and technologically-mediated health care injuries. *St Louis Univ Law J* 2002; 45:37–59.
60. Osheroff JA, Pifer EA, Sittig DF, Jenders RA, Teich JM. Clinical decision support implementers’ workbook. Chicago, HIMSS, 2004. Available from: www.himss.org/cdsworkbook. Second edition: Improving outcomes with clinical decision support: an implementers’ guide; HIMSS, 2005.
61. Brannigan VM, Dayhoff RE. Medical informatics. The revolution in law, technology and medicine. *J Leg Med* 1986;7:1–53.
62. Sim I, Gorman P, Greenes RA, Haynes RB, Kaplan B, Lehmann H, Tang PC. Clinical decision support for the practice of evidence-based medicine. *J Am Med Inform Assoc* 2001;8:527–534.
63. Kittelson B. Spotlight: CPR systems. *Healthc Inform* 2002;18:41–47.
64. Marietti C. The 2002 Healthcare Informatics 100. *Healthc Inform* 2002;18:33–64.

65. Poikonen J. Arden syntax: the emerging standard language for representing medical knowledge in computer systems. *Am J Health Syst Pharm* 1997;54: 281–284.
66. Hsieh TC, Kuperman GJ, Jaggi T, et al. Characteristics and consequences of drug allergy alert overrides in a computerized physician order entry system. *J Am Med Inform Assoc* 2004;11:482–491.
67. Berner ES, Maisiak RS, Heudebert GR, Young KR Jr. Clinician performance and prominence of diagnoses displayed by a clinical diagnostic decision support system. *AMIA Annual Symp. Proc* 2003:76–80.
68. Friedman CP, Gatti GG, Franz TM, et al. Do physicians know when their diagnoses are correct? Implications for decision support and error reduction. *J Gen Intern Med* 2005;20:334–339.
69. Miller RA, Schaffner KF, Meisel A. Ethical and legal issues related to the use of computer programs in clinical medicine. *Ann Intern Med* 1985;102:529–536.
70. Fried BM, Zuckerman JM. FDA regulation of medical software. *J Health Law* 2000;33:129–140.
71. Miller RA, Gardner RM. Recommendations for responsible monitoring and regulation of clinical software systems. *J Am Med Inform Assoc* 1997;4: 442–457.
72. Osheroff JA, Teich JM, Middleton BF, Steen EB, Wright A, Detmer DE. A roadmap for national action on clinical decision support. For AMIA as a work product for ONC contract HHSP23300500877P. June 13, 2006. Available from: <http://www.amia.org/inside/initiatives/cds/cdsroadmap.pdf>. Accessed on July 8, 2006.

2

Mathematical Foundations of Decision Support Systems

S. ANDREW SPOONER

Many computer applications may be considered to be clinical decision support systems. Programs that perform Medline searches or check drug interactions do support decisions, but they are not “clinical decision support systems” in the usual sense. What we usually mean by a clinical decision support system (CDSS) is a program that supports a reasoning task carried out behind the scenes and based on clinical data. For example, a program that accepts thyroid panel results and generates a list of possible diagnoses is what we usually recognize as a diagnostic decision support system, a particular type of CDSS. General purpose programs that accept clinical findings and generate diagnoses are typical diagnostic decision support systems. These programs employ numerical and logical techniques to convert clinical input into the kind of information that a physician might use in performing a diagnostic reasoning task. How these numerical techniques work is the subject of this chapter.

Essential to the understanding of CDSS is familiarity with the basic principles of logic and probability. A brief review of these areas is offered first, followed by a description of a general model of CDSS, which will help in understanding how some CDSS perform reasoning tasks. Exceptions to the model will round out this discussion of the mathematical foundations of CDSS.

Review of Logic and Probability

Set Theory

A brief review of basic concepts in set theory is helpful in understanding logic, probability, and many other branches of mathematics. A *set* is a collection of unique objects. For example, the major Jones criteria¹ for rheumatic fever is a set:

JONES-CRITERIA-MAJOR = {carditis, migratory polyarthritis, erythema marginatum, chorea, subcutaneous nodules}

Likewise, the minor criteria make a set:

JONES-CRITERIA-MINOR = {fever, arthralgia, elevated acute phase reactants, prolonged P-R interval on electrocardiogram}

To complete our description of the Jones criteria, we need a third set:

GROUP-A-STREP-EVIDENCE = {positive culture, positive rapid antigen, antibody rise or elevation}

To apply the Jones criteria, one compares the patient's findings with the items in the various sets above. A patient is highly likely to have rheumatic fever if there is evidence of group A streptococcal infection and the patient has two major criteria or one major and two minor criteria.

Each *element* or *member* of the set is distinguishable from the others. A *subset* is any collection of elements of a known set. Using the first of the criteria above, a patient must have a subset of clinical findings containing at least two of the elements of JONES-CRITERIA-MAJOR to meet the Jones criteria for rheumatic fever. If a patient has the clinical findings:

FINDINGS = {migratory polyarthritis, chorea, subcutaneous nodules}

then we say that FINDINGS is a subset of JONES-CRITERIA-MAJOR, or, in set terminology:

$\text{FINDINGS} \subseteq \text{JONES-CRITERIA-MAJOR}$

The *cardinality* or *size* of a set is simply the number of elements in the set. For our two examples, the cardinalities (written by placing a vertical bar before and after the symbol for the set) are:

$|\text{FINDINGS}| = 3$

$|\text{JONES-CRITERIA-MAJOR}| = 5$

The basic set operations are *intersection* and *union*. The intersection of two sets is the set of elements the two sets have in common. For example, if there is a patient with the following set of clinical findings:

CLINICAL-FINDINGS = {heart murmur, migratory polyarthritis, chorea, subcutaneous nodules, cough}

then the intersection of this set and JONES-CRITERIA-MAJOR is written:

$\text{CLINICAL-FINDINGS} \cap \text{JONES-CRITERIA-MAJOR}$

It is easy to see that the intersection of these two sets is simply the set FINDINGS. The union of two sets is the set of all elements that belong to either set. Since, by definition, a set's elements must be distinguishable from one another, the set resulting from the union of our patient's findings and the Jones major criteria is written:

CLINICAL-FINDINGS \cup JONES-CRITERIA-MAJOR = {heart murmur, migratory polyarthritis, chorea, subcutaneous nodules, cough, carditis, erythema marginatum, chorea}

Anyone who has done a Medline search in which two sets of literature citations are combined has performed these set operations; the AND function in Medline is like set intersection, and the OR function is like set union.

Diagnostic criteria like the Jones criteria are good examples of how sets can be used to represent diagnostic rules. The full Jones criteria, represented in set theoretical terminology, might read like this (assuming we have sets JONES-CRITERIA-MINOR and GROUP-A-STREP-EVIDENCE as described at the beginning of this section):

If CLINICAL-FINDINGS is the set of a given patient's symptoms, signs, and laboratory test results, then the patient is highly likely to have rheumatic fever if either of two conditions are met:

1. $|\text{CLINICAL-FINDINGS} \cap \text{JONES-CRITERIA-MAJOR}| \geq 2$
and
 $|\text{CLINICAL-FINDINGS} \cap \text{GROUP-A-STREP-EVIDENCE}| \geq 1$
2. $|\text{CLINICAL-FINDINGS} \cap \text{JONES-CRITERIA-MAJOR}| = 1$
and
 $|\text{CLINICAL-FINDINGS} \cap \text{JONES-CRITERIA-MINOR}| \geq 2$
and
 $|\text{CLINICAL-FINDINGS} \cap \text{GROUP-A-STREP-EVIDENCE}| \geq 1$

There are other set operations besides union and intersection. For example, the phenomenon of *set covering* has applications in decision making. A cover of a set is a set of subsets in which each element of the covered set appears at least once as a member of one of the sets in the cover set. An example makes this definition clearer. Suppose you were asked to recommend a list of antibiotics for your hospital's emergency department. Your objective is to stock the minimum number of antibiotics that will be effective for 95% of the pathogenic organisms you've found in cultures at your hospital. For the sake of simplicity, suppose that there are six pathogens, each designated by a letter, which account for 95% of the infections seen in your hospital. You might represent this set of pathogens as:

PATHOGENS = {A, B, C, D, E, F}

You have the following set of antibiotics from which to choose:

ANTIBIOTICS = {A-Cillin, B-Cillin, C-Cillin, D-Cillin, E-Cillin, F-Cillin}

Each antibiotic is described by the set of pathogens for which that antibiotic is effective. Here is a list of your antibiotics, with their covered pathogen sets (each of which is a subset of PATHOGENS):

- A-Cillin = {A, C}
- B-Cillin = {A, B, E}
- C-Cillin = {C, D, E}
- D-Cillin = {F}
- E-Cillin = {B, D, F}
- F-Cillin = {E}

What you seek is a set cover of the set PATHOGENS; in other words, you want to pick a set of antibiotics which contains at least one antibiotic that is effective for each pathogen. It's clear that all six antibiotics taken together make a set cover, but your job is to find the minimum number of antibiotics that will get the job done. Casual inspection shows that the set {A-Cillin, E-Cillin, F-Cillin} does the job as a set cover, in that at least one antibiotic in that set is effective for each one of the pathogens in PATHOGENS.

There are many other set operations which can be applied to real-world decision problems, but the brief introduction presented here should suffice to illuminate the concepts presented in this book. Generally speaking, sets are used to formalize logical operations in a way that a machine—usually a computer—can understand.

Before we leave the topic of sets, fuzzy sets are worth a brief mention. Under conventional principles of set theory, an element is either a member of a set or it isn't. Heart murmur, for example, is definitely not a member of the set JONES-CRITERIA-MAJOR. Under *fuzzy set theory*, membership in a set is not an all or nothing phenomenon. In a fuzzy set, an element is a member of the set with a certain probability; e.g., cough is a member of the set COLD-SYMPTOMS with a probability of 80% (a 4 out of 5 chance). Fuzzy set theory has created new ways of looking at sets and new methods for applying set theory to solve decision-making problems: fuzzy logic.^{2,3} Fuzzy logic has been used to tackle decision-making problems in which uncertainty plays a role.

Boolean Logic

Anyone who has performed a search of the medical literature using the Medline system has used logic. When referring to common logical operations like combining two sets of literature citations using AND or OR, we often refer to these operations as “Boolean” logic, in honor of George Boole (1815–1864), a British mathematics professor who published seminal works on formal logic. Indeed, Medline is not a bad way to learn about Boolean algebra, since its connection to set theory is made so clear by the sets of literature citations that we manipulate in that system.

Suppose we have performed two literature searches. The result of one search, set A, represents all the literature citations in the Medline database that relate to rheumatoid arthritis. Set B consists of all the literature

citations on immune globulin. By asking the Medline program to give us a new set that is the result of combining A and B using the AND operator, we have a new set, C, that contains literature citations on the use of immune globulin in rheumatoid arthritis. When we combine two sets of citations using the AND function of our Medline program, we are asking the computer to give us all citations that appear in both sets. This corresponds roughly to the English use of the word *and*.

The word OR in Boolean logic has a slightly different meaning than in English. In everyday usage, *or* usually has an exclusive meaning; the statement, “You may opt for chemotherapy or radiation therapy,” usually means that one may have one or the other therapy, but not both. The Boolean OR is different. If one were to perform another pair of Medline searches, this time for all articles that have asthma as a keyword (set A) and those that mention “reactive airway disease” in the text of the abstract (set B), one could combine sets A and B with the OR function to get a comprehensive set of citations on asthma. Because the OR function takes all citations that appear in one or both of sets A and B, the OR function is said to be *inclusive*.

There are other Boolean operators, like XOR (exclusive OR: “either A or B but not both”) and NAND (“A and not B”), but AND and OR are the basic operators with which we are familiar.

How is Boolean logic used in CDSS? The mathematical subjects of statement logic and predicate logic give us formal definitions of how statements can be combined to produce new conclusions. For example, consider the following statements:

1. Urine cultures with colony counts of 10,000 or more are considered positive if they are obtained by bladder catheterization.
2. This patient’s urine culture shows more than 10,000 colonies of *E. coli*.
3. All patients with positive urine cultures should be treated for urinary tract infections.

The statements can be combined intuitively, without the use of formal mathematics, into the conclusion:

This patient needs to be treated for a UTI.

The logic that gave us the conclusion so easily, comes from our medical intuition, but computers have no intuition. They must be programmed to generate even the most obvious conclusions. To understand logic as it is implemented on a computer, one must understand the basics of predicate logic and deductive reasoning.

The above example about UTIs is a sloppy instance of a syllogism. A syllogism is a form of deductive reasoning consisting of a major premise, a minor premise, and a conclusion. The premises are combined, using rules of predicate logic, into a conclusion. For example, a syllogism in a ventilator management decision support system might be:

Major Premise: All blood gas determinations that show carbon dioxide to be abnormally low indicate an over-ventilated patient.

Minor Premise: The current patient's carbon dioxide is abnormally low.

Conclusion: Therefore, the current patient is over-ventilated.

Again, this conclusion is obvious, but by representing the above syllogism using symbols, where the symbol Low-CO_2 represents the state of abnormally low carbon dioxide and the symbol OVERVENTILATED represents the state of an over-ventilated patient, the syllogism looks more computer-friendly:

Major Premise: $\text{Low-CO}_2 \Rightarrow \text{OVERVENTILATED}$

Minor Premise: Low-CO_2

Conclusion: OVERVENTILATED

Extending this example, suppose we have another statement in our CDSS that over-ventilation should cause a high rate alarm to sound (we can represent this by the symbol HIGH-RATE-ALARM), then we can construct the syllogism:

Major Premise: $\text{Low-CO}_2 \Rightarrow \text{OVERVENTILATED}$

Minor Premise: $\text{Over-ventilated} \Rightarrow \text{HIGH-RATE-ALARM}$

Conclusion: $\text{Low-CO}_2 \Rightarrow \text{HIGH-RATE-ALARM}$

Thus, we have generated a new rule for the system, where the intermediate state of over-ventilation is bypassed. This simplification of two rules into a new one may or may not help our understanding of the system, but the results the system gives are the same: a low carbon dioxide value sets off the high rate alarm. One can imagine how large sets of rules can be combined with each other to reduce complex reasoning tasks to simple ones.

The syllogism above is an example of rule chaining, where two rules are chained together to form a new conclusion. Specifically, the simple system outlined above is a *forward-chaining deduction system*, because the system starts with *if* statements and moves to a *then* statement. In real life, though, we often start with the “then” portion of a logical rule. For instance, consider the clinical rule:

If your patient has asthma, then give an influenza immunization each fall.

There are many other rules in real clinical practice with the same “then” portion (“give a flu vaccine”). The question a clinician might ask is not “Does this patient have asthma? If so, I should give a flu shot,” but more likely the question would be simply “Does this patient need a flu shot?” We start with the “then” portion of this set of flu shot rules. A *backward-chaining deduction system* does this—it starts with the “then” end of a set of rules and works backwards to answer questions based on its rule set. In the flu shot example, a backward-chaining system would start with the “Does this patient need a flu shot” question and immediately learn that

the diagnosis of asthma would cause this rule to be satisfied. The system might then ask the user or query a clinical database about the presence of this diagnosis.

An example of a backward-chaining deduction system in medicine is the MYCIN system developed at Stanford.⁴ MYCIN's domain was the selection of antibiotics for the treatment of bacterial infections based on clinical and microbiological information. An example of a forward-chaining system in medicine is Germwatcher, developed at Barnes Hospital in St. Louis.⁵ Germwatcher uses as its rules the Centers for Disease Control and Prevention's National Nosocomial Infections Surveillance System.⁶ Using a computer program which helps implement a forward-chaining reasoning system called CLIPS (C Language Integrated Production System, Software Technology Branch, National Aeronautics and Space Administration, Johnson Space Center, Houston, TX), expert system shell Germwatcher works in a large hospital microbiology laboratory to identify nosocomial infections early from culture data.

CDSS that use logic like the simple ventilator-management system above have limited application, since the range of truth encompassed by this logical system includes only true (e.g., the High Rate alarm needs to be sounded) or false (e.g., the High Rate alarm does not need to be sounded). Not many applications in medicine can be reduced to such simple truths. There may be situations where the High Rate alarm might not always have to be sounded for a low carbon dioxide reading (e.g., for a head injury patient who needs low carbon dioxide to preserve cerebral blood flow). To accommodate these situations, it would be helpful if the response from the system were something like "the high rate alarm should probably be sounded." Such a system would then need to be able to handle probabilities, as well as certainties, which most CDSS do. MYCIN, for example, reports its conclusions in terms of their likelihood. The next section covers basic concepts of probability.

Probability

Everyday medical practice contains many examples of probability. We often use words such as probably, unlikely, certainly, or almost certainly in all conversations with patients. We only rarely attach numbers to these terms, but computerized systems must use some numerical representation of likelihood in order to combine statements into conclusions.

Probability is represented numerically by a number between 0 and 1. Statements with a probability of 0 are false. Statements with a probability of 1 are true. Most statements from real life fall somewhere in the middle. A probability of 0.5 or 50% are just as likely to be true as false. A round, opacified area seen in the lungs on a chest radiograph is probably pneumonia; one might assign a probability of 0.8, or 80%, (a 4 in 5 chance) to this statement. Based on the high probability of pneumonia, one might elect

to treat this condition without performing further testing—a lung biopsy, perhaps—that would increase the probability of pneumonia to greater than 80%. We are accustomed to accepting the fact that our diagnoses have a certain probability of being wrong, so we counsel patients about what to do in the event (we might use the term “unlikely event”) that things don’t work out in the expected way.

Probabilities can be combined to yield new probabilities. For example, the two statements:

$$\Pr(\text{diabetes}) = 0.6$$

$$\Pr(\text{hypertension}) = 0.3$$

mean that the probability of diabetes is 0.6, or 60%, (3 in 5 chance), and the probability of hypertension is 0.3, or 30%, (3 in 10 chance). We have not specified the clinical context of these statements, but suppose these probabilities applied to a particular population. Suppose further that the two conditions are independent; that is, the likelihood of patients having one disease is unaffected by whether they have the other (not always a safe assumption!). If we then want to know what the probability is, of finding a patient in our specified population with both diseases, we simply multiply the two probabilities (0.6 and 0.3) to get 0.18, or 18%. If the two clinical conditions are not independent, (e.g., pulmonary emphysema and lung cancer) then we cannot combine the probabilities in such a simple, multiplicative manner. This is much like the AND function in Medline or the intersection function as applied to sets.

The familiar “OR” function from our Medline program also has a mathematical meaning in combining probabilities. If we wanted to know how many patients in the above example had diabetes *or* hypertension (remember: this would also include those with both diseases in the usual mathematical sense of *or*), we would compute:

$$\Pr(\text{diabetes OR hypertension}) = \Pr(\text{diabetes}) + \Pr(\text{hypertension}) - \Pr(\text{diabetes AND hypertension}).$$

The last term in the above equation we already know to be $0.6 \times 0.3 = 0.18$, so:

$$\Pr(\text{diabetes OR hypertension}) = 0.6 + 0.3 - 0.18 = 0.72.$$

Conditional probability is another type of probability often used in medicine. A conditional probability is the probability of an event (or the probability of the truth of a statement) *given the occurrence of another event* (or the truth of another statement). The most familiar case of conditional probability in medicine arises in the interpretation of diagnostic tests. For example, the probability of pneumonia given a round density on a chest radiograph is what we need to know in interpreting that diagnostic test if it is positive. In mathematical notation, this conditional probability is written this way:

$\Pr(\text{Pneumonia} \mid \text{Round Density on CXR})$.

One reads this notation, “The probability of pneumonia given a round density on chest radiograph.” This notation is convenient in the explanation of Bayes’ rule, which is the cornerstone of the logic in many decision support systems.

Bayes’ Rule

If we have a patient with jaundice, how likely is it that he has hepatitis? Written another way, we seek to learn:

$\Pr(\text{hepatitis} \mid \text{jaundice})$,

which is read as “the probability of hepatitis given the presence of jaundice.” We may not have this probability at our fingertips, but we might be able to find a slightly different probability more easily:

$\Pr(\text{jaundice} \mid \text{hepatitis})$,

which is, simply, the probability of jaundice given the presence of hepatitis. The latter probability could be found by studying a series of patients with proven hepatitis (it would be easy to get this data by looking up diagnosis codes in the medical records department) and computing the percentage of these patients who present with jaundice. However, this does not directly answer our original question. Bayes’ rule allows us to compute the probability we *really* want— $\Pr(\text{hepatitis} \mid \text{jaundice})$ —with the help of the more readily available number $\Pr(\text{jaundice} \mid \text{hepatitis})$. Bayes’ rule⁷ is simply this:

$$\Pr(\text{hepatitis} \mid \text{jaundice}) = \frac{\Pr(\text{hepatitis}) \times \Pr(\text{jaundice} \mid \text{hepatitis})}{\Pr(\text{jaundice})}$$

Notice that to solve this equation, we need not only $\Pr(\text{jaundice} \mid \text{hepatitis})$, but $\Pr(\text{hepatitis})$ —the probability of hepatitis independent of any given symptom—and $\Pr(\text{jaundice})$ —the probability of jaundice independent of any particular disease. These two independent probabilities are called *prior probabilities*, since they are the probabilities prior to the consideration of other factors.

The derivation of Bayes’ rule is very simple. We already know that the probability of any two events occurring simultaneously is simply the product of their individual probabilities. For example, the joint probability we already computed of diabetes and hypertension in a hypothetical population was:

$$\begin{aligned} \Pr(\text{diabetes AND hypertension}) &= \Pr(\text{diabetes}) \times \Pr(\text{hypertension}) \\ &= 0.6 \times 0.3 = 0.18. \end{aligned}$$

We were free to multiply these together, because in our hypothetical population, the likelihood of one disease occurring in an individual was independent of the other. In other words:

$$\Pr(\text{hypertension}) = \Pr(\text{hypertension} \mid \text{diabetes})$$

and

$$\Pr(\text{diabetes}) = \Pr(\text{diabetes} \mid \text{hypertension}).$$

In this population, one's chance of having one disease is unaffected by the presence of the other disease.

In medicine, we are often faced with the question of the likelihood of two interrelated events occurring simultaneously in a patient. The case of a diagnostic test and the disease it is supposed to test for is a good example: what is the probability of an abnormal chest radiograph and pneumonia occurring in the same patient simultaneously? The question asks for this probability:

$$\Pr(\text{pneumonia AND abnormal CXR}).$$

Can't we simply find out what the incidence of pneumonia in the population is, and multiply it by the incidence of abnormal chest radiographs in the population? A moment's reflection should show that this simple calculation is not sufficient. For example, if the incidence of pneumonia is 1 in 1000, and the incidence of abnormal chest radiograph is 1 in 100, then the erroneous probability would be computed:

$$\begin{aligned} \text{WRONG: } \Pr(\text{pneumonia AND abnormal CXR}) &= \frac{1}{1000} \times \frac{1}{100} \\ &= 0.00001 = 0.001\% \end{aligned}$$

This does not fit with our clinical intuition very well, since we know that people with pneumonia tend to have abnormal chest films. Our intuition says that the probability of the two events occurring together should be pretty close to the probability of having pneumonia alone, since a majority of those patients will have abnormal chest films. What we *really* need to compute is this:

$$\Pr(\text{pneumonia AND abnormal CXR}) = \Pr(\text{pneumonia}) \times \Pr(\text{abnormal CXR} \mid \text{pneumonia}).$$

This is the probability of pneumonia multiplied by the probability of an abnormal chest radiograph given that pneumonia exists. If we take $\Pr(\text{abnormal CXR} \mid \text{pneumonia})$ to be 90%, then the computation matches our intuition much better.

In general, for any two events A and B:

$$\Pr(A \text{ AND } B) = \Pr(A) \times \Pr(B \mid A)$$

and

$$\Pr(B \text{ AND } A) = \Pr(B) \times \Pr(A | B).$$

But since $\Pr(A \text{ AND } B)$ must surely equal $\Pr(B \text{ AND } A)$, we can say that the right-hand sides of the equations above are equal to each other:

$$\Pr(A) \times \Pr(B | A) = \Pr(B) \times \Pr(A | B)$$

Rearranging this equation, we have Bayes' rule:

$$\Pr(A | B) = \frac{\Pr(A) \times \Pr(B | A)}{\Pr(B)}$$

At an intuitive level, we use Bayes' rule when making seat-of-the-pants estimates of disease probability in patients. For example, if we designate hepatitis by A and jaundice by B , and there were an on-going epidemic of hepatitis (i.e., $\Pr(A)$ was high), then our index of suspicion for hepatitis in a jaundiced person would be increased. Likewise, if the likelihood of jaundice due to other causes was high (i.e., $\Pr(B)$ was high), then our estimation of the probability of hepatitis as a specific diagnosis would be lowered. Similarly, if jaundice were pathognomonic of hepatitis (i.e., $\Pr(A | B)$ was 1 or near to it), then our hepatitis diagnosis would be greatly increased. By using numerical estimates of the probability of diseases, findings, and conditional probabilities, Bayes' rule can help make medical decisions.

One might imagine a simple CDSS in which one enters a single symptom and receives the probability of the presence of a disease given that symptom. A problem arises when one wishes to get disease probabilities given multiple symptoms. The number of data points needed to do Bayesian calculations on multiple simultaneous symptoms is huge. For example, in a system which handles only single symptoms, if one had a database of 1000 symptoms and 200 diseases, one would need to create $1000 \times 200 = 200,000$ conditional probabilities, 1000 symptom probabilities, and 200 disease probabilities, for a total of about 200,000 numbers. Since most of these numbers are 0 (many symptoms are unrelated to many diseases), this may be a reasonable amount of numbers to collect into a knowledge base. When one starts considering the probabilities needed to do computations on two simultaneous symptoms, this number climbs from 200,000 to about 200,000,000! If one wanted to design a system that could handle the very realistic situation of 5 or 6 simultaneous symptoms, estimating the number of numbers needed to support the calculation would be intractable. Modifying the system to handle multiple simultaneous "diseases" adds even more to the complexity. Only after making the simplifying assumption that most disease findings are independent of one another⁸ do many CDSS use Bayesian approaches. One such system, *Iliad*,⁹ profitably employed this assumption.

Informal Logic

Even if we create a reasoning system that follows all the rules of logic and probability, it would be difficult to come up with all the numbers that must be assigned to each event in even a small clinical database. Many successful CDSS have circumvented this difficulty by employing informal rules of logic to accomplish the reasoning task, without creating an intractable data-gathering task. In the early development of one of the most famous CDSS, MYCIN,^{4,10} the creators of the system developed their own logic system (heuristic) that made intuitive sense. This system employed “certainty factors” which ranged from -1 (false) to $+1$ (true). A certainty factor of 0 indicated no belief in either direction in the statement’s veracity. In combining several statements with the AND function into a single combined statement in MYCIN, one simply takes the minimum certainty factor of all the statements as the certainty factor of the combined statement. This makes a certain intuitive sense: we cannot be any more certain of an AND statement than we are of the least certain part. Later development of the MYCIN project showed a sound probabilistic basis for the certainty factor rules, but the point here is that sometimes cutting mathematical corners can still yield a useful system. In both the QMR¹¹ and DXplain¹² CDSS, there is a database of diseases and findings (a finding is an item from the history, physical examination, laboratory data, or radiographic data). Each disease is defined by a particular set of findings. Each disease-finding relationship is assigned a frequency (of the finding among people with the disease) and an evoking strength (of how strongly a finding would evoke the possibility of a disease) on an ordinal scale (1–5 for frequency; 0–5 for evoking strength). These two factors make intuitive sense, and the system works, but the manipulation of these factors within these systems is very different from the formal algebra of logic and probability.

The General Model of Knowledge-Based Decision Support Systems

There are similarities between physician and CDSS reasoning, although a CDSS might arrive at a similar conclusion to a physician without employing the same model of reasoning. Physicians do use some probabilistic information when they make decisions. For instance, a physician might make a diagnosis of influenza more often during the winter when influenza is more prevalent (probable) than in the summer. However, physicians use this information in informal ways; in other words, they do not actually use numbers in formulas to make diagnostic decisions.^{13–15} Another feature of real-life clinical decision making is that physicians do not require complete information to make a decision. Most doctors are comfortable making decisions based on incomplete or contradictory information.¹⁶ In contrast,

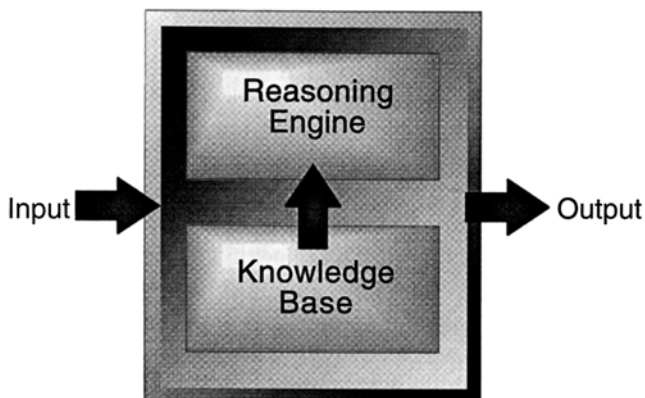


FIGURE 2.1. A general model of a clinical diagnostic decision support system.

CDSS rely on well defined numerical techniques to do their reasoning, and they do require sufficient information to complete their formulae. While physicians can fall back on their knowledge of pathophysiology, CDSS are not well suited to situations in which hard data are unknown. To understand how these systems operate, and under what conditions they are best used, it is important to appreciate a general model of CDSS.

Figure 2.1 shows a general model of CDSS. There is input to the system and output from it. The CDSS has a reasoning (inference) engine and a knowledge base. Understanding these basic components provides a useful framework for understanding most CDSS and their limitations. There are systems which do not follow this model which will be discussed briefly later in this chapter and in Chapter 3 in more detail.

The user supplies input appropriate to the system (i.e., terms from the system's controlled vocabulary to represent clinical data), and the system supplies output (e.g., a differential diagnosis). The reasoning engine applies formal or informal rules of logic to the input and often relies on additional facts encoded in the system's knowledge base. The knowledge base is the compilation of the relationships between all of the diseases in the system and their associated manifestations (e.g., signs, symptoms, laboratory and radiographic tests). Maintaining the knowledge base in such systems is the most significant bottleneck in the maintenance of such systems, since the knowledge base needs to be expanded and updated as medical knowledge grows.

Input

The manner in which clinical information is entered into the CDSS (user interface) varies from system to system, but most diagnostic systems require the user to select terms from its specialized, controlled vocabulary.

Comprehension of natural language has been an elusive goal in the development of CDSS. While it would be highly desirable to be able to speak or type the query “What are the diagnostic possibilities for a four-year-old child with joint swelling and fever for a month,” most who have used such systems are accustomed to the task of reformatting this question in terms the particular CDSS can understand. We might, for example, break the above query into components:

- Age: 4 years
- Gender: unspecified
- Symptom: joint swelling
- Duration: 1 month
- Time course: unknown

This breakdown of the original query might work on one system, but another system might demand that we break it down another way:

- Age: less than 12 years
- Finding: arthritis

Notice that the second description describes the age in vague terms, and it forces us to eschew joint swelling for the more specific term arthritis (usually defined as joint pain, redness, warmth, and swelling). In the vocabulary of the program, the age of four years (as opposed to 10 years) is unimportant, and joint swelling, without other signs of inflammation, is undefined.

Any physician who has assigned diagnostic and procedural codes in billing systems understands the limitations of controlled vocabularies. In a CDSS, it is common for the user’s input to be restricted to a finite set of terms and modifiers. How well the system works in a given clinical situation may depend on how well the system’s vocabulary matches the terms the clinician uses. CDSS take a variety of terms, called findings, which encompass items from the medical history, physical examination, laboratory results, and other pieces of clinical information. What constitutes a valid finding in a given program is entirely up to the program; there is no “standard” set of findings for all CDSS. For general purpose CDSS, items from the history and physical examination are going to be the findings. In specialized domains, e.g., an arterial–blood–gas expert system, the input vocabulary will be entirely different and much more restrictive.

Entering “chest pain” as a finding in a CDSS may be insufficient to capture the essence of the symptom. “Chest pain radiating to the left arm” may be sufficient, but usually there are pertinent temporal factors related to symptoms that are difficult to express in a controlled vocabulary. For example, “sudden onset, 20 minutes ago, of chest pain radiating to the left arm” has a very different meaning from “five-year history of continuous chest pain radiating to the left arm.” While CDSS often include a vocabulary of severity and location modifiers, temporal modifiers are more difficult to build into a system, since minute changes in the timing of onset and

duration can make a big difference in the conclusion the system reaches. Some CDSS make simplifying assumptions about broad categories of timing (acute, subacute, chronic) to aid in the temporal description of findings. Although users may experience frustration in being unable to enter temporal information, the research is equivocal in its impact.

One solution to the problem of temporal modeling in CDSS is to use an explicit model of time, in which the user is asked to specify intervals and points in time, along with temporal relationships between events (e.g., event A occurred before event B), in order to drive a temporal reasoning process within the CDSS. Clearly, this complicates the matter of entering data (to say nothing of programming the system). A simpler approach is to model time implicitly. In implicit time,¹⁷ temporal information is built into the data input elements of the CDSS; no special temporal reasoning procedures are required. For example, one input item could be “history of recent exposure to strep.” By joining the concept “history of” with the concept of a particular bacterial pathogen, one successfully abstracts the temporal nature of this finding, which would be pertinent in the diagnosis of rheumatic fever or post-streptococcal glomerulonephritis. Note that no explicit definition of “recent” is part of this representation; if for some reason one needed to distinguish infection 2 weeks ago from infection 3 months ago, this abstraction would not suffice. Thus, there is a disadvantage to this simplification. Nonetheless, CDSS which use implicit temporal abstractions seem to perform well for time-sensitive clinical cases.

Inference Engine

There are many ways of programming an inference engine. The inference engine is the portion of the CDSS that combines the input and other data according to some logical scheme for output. Users of the system do not usually know—or need to know—how the engine works to achieve the results.

One such scheme for an inference engine is the Bayesian network. Recall that Bayes’ rule helps us express conditional probabilities—the likelihood of one event given that another has occurred. A Bayesian network is a way to put Bayes’ rule to work by laying out graphically which events influence the likelihood of occurrence of other events. Figure 2.2 shows a Bayesian network for the diagnosis of pneumonia.

The arrows in the diagram indicate all of the conditional relationships between findings and diagnoses. Note that the symptoms listed are not necessarily independent; since febrile patients are often tachypneic, even in the absence of lung disease, one cannot say the two are as independent as Bayesian reasoning requires. Conceptually, this network simply states that the diagnosis of pneumonia is supported by the presence of three symptoms. The strength of association—that is, how strongly pneumonia is suggested by each of the three symptoms—varies with each symptom–disease

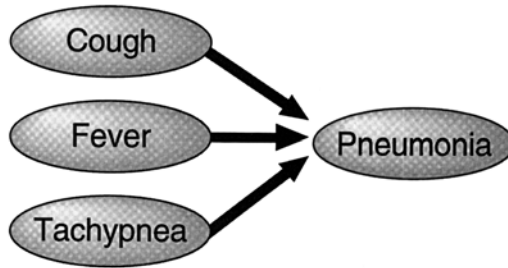


FIGURE 2.2. A Bayesian network for the diagnosis of pneumonia.

pairing. By “activating” all three nodes (cough, fever, and tachypnea) the probability of pneumonia is maximized. Of course, each of these three nodes might be tied to other disease states in the knowledge base (like lung cancer or upper respiratory infection).

Bayesian networks can be complex, but their usefulness comes from their ability to represent knowledge in an intuitively appealing way. Inference engines that operate on the basis of a network simply adjust probabilities based on simple mathematical relationships between nodes in the network. Iliad,⁹ a general CDSS, is one such program that is built on Bayesian reasoning, and whose reasoning engine can be described as a Bayesian network. Mammonet¹⁸ is a mammography CDSS built on a Bayesian network whose nodes include quality of the breast mass, age at menarche, age of the patient, type of calcification of the mass, and other findings apt to be present in the evaluation of a mammogram.

Production rule systems are another method of programming an inference engine. The rules of predicate logic dictate the functioning of such an engine as it combines statements to form new conclusions. MYCIN, described earlier, uses a production rule system. Production rules are an intuitively attractive way to start thinking about CDSS, since so much of the care physicians give in daily practice follows certain well known rules (e.g., giving patients with asthma an influenza vaccine each year). Other CDSS using production rules include IMM/Serve,¹⁹ a rule-based immunization decision-making program developed at Yale University and Hepaxpert (MED-EXPERT Data Systems Ltd., Vienna, Austria), a rule-based hepatitis serology expert system. Production rule systems can get quite complicated beyond narrow domains.

An appealing solution to the problem of constructing inference engines in a clinical setting is to develop a cognitive model of actual clinical reasoning. In other words, one could study the reasoning that a physician uses and attempt to create a computerized version of that cognitive task. Workers in the field of artificial intelligence, in modeling human cognition, have developed the notion of “frames” or schemes, as a reasonable

cognitive model. A frame consists of a set of “slots” into which fit details of a particular kind of information. For example, a disease frame may have a slot for etiologic agent and time course. Frames can be used to construct a semantic network model of the world, which may then be searched for answers to questions based on a particular situation. One such application of frames in medicine is the criterion-table method of diagnosing diseases like rheumatoid arthritis or Kawasaki disease. By applying a list of criteria, physicians can classify patients by diagnosis. The AI/Rheum system²⁰ employs this familiar device in an inference engine that can be used outside its original domain of rheumatologic diseases.

One important aspect of inference engines is their independence from their knowledge base. Since CDSS take a great deal of time to develop, reusability has been a focus of research.²¹ Theoretically, one should be able to take any inference engine and apply it in any domain. In reality, a given inference engine is developed with a particular domain in mind, and its use does not move from that domain.

Knowledge Base

For CDSS to work, they must possess some form of medical knowledge. Obviously, the method of encoding this knowledge must match the inference engine design. For example, a CDSS based on a Bayesian network must contain probabilities—prior, conditional, and posterior—of diseases and findings. A big obstacle to building such a knowledge base is that many relevant probabilities are not known. While the medical literature can surely help with this task, and CDSS developers use the literature to varying degrees in building their knowledge bases, knowledge base developers must resort to estimates of probabilities, based on the clinical judgment of experts, to fill in the needed numbers. Unfortunately, physicians can exhibit markedly variable behavior in supplying such numbers,²² and probabilities can vary from situation to situation, even with the same disease entities (e.g., variations in disease prevalence with different populations).

Once one creates a knowledge base and populates it with some amount of data, the next task is to create a way to maintain it. Since many CDSS begin as funded academic research projects, it is no wonder that development of their knowledge bases often halts after the grant funds cease. Since knowledge base maintenance takes a tremendous amount of time, and since the market for some CDSS is rather small, many CDSS become too expensive to maintain. The knowledge-acquisition bottleneck²³ has been recognized as a problem in CDSS research.

Output

The output of CDSS is usually in the form of a list of possibilities, ranked in some order of probability. Sometimes probability is not the only

criterion on which results are evaluated; for example, in the DXplain output, diseases which are not necessarily very likely, but whose misdiagnosis would be catastrophic, are flagged with a special disease-importance tag to call attention to the possibility.¹² Very often, physicians are not interested in the most likely diagnosis from a CDSS; for experienced physicians, the most likely diagnosis is obvious. It is the less likely diagnosis that one might fail to consider that interests physicians in CDSS, yet clearly it is difficult to draw the line between the rare and the ultra-rare.

Nonknowledge-Based Systems

The systems discussed so far have been knowledge-based in the sense that an expert must expressly encode medical knowledge into numerical form for the systems to work. The knowledge-based systems cannot simply “learn” how to do the reasoning task from modeling human experts; the human expert must put the knowledge into the system explicitly and directly.

Neural Networks

There are systems that can learn from examples. Neural networks are the most widely recognized of these types of systems, and there are regular reports in the medical literature on their use in diverse fields.²⁴⁻²⁷

Artificial neural networks are constructed in a fashion similar to biological neural networks. Neuron bodies (“nodes”) are connected to one another by axons and dendrites (“links”). Nodes may be turned on or off, just as a biological neuron can be in an activated or inactivated state. Activation of a node causes activation of a signal on a link. The effect of that signal depends on the weight assigned to that link. In most learning neural networks, some nodes are input nodes and some are output nodes. In the CDSS context, the input nodes would be findings and the output nodes would be possible diseases. To understand how a neural network might work, consider the problem of determining whether a person with a sore throat has streptococcal infection (as opposed to a harmless viral infection). There are many input nodes to this decision, and perhaps two output nodes, strep infection and viral infection. By presenting to a neural network many thousands of cases of sore throat (where the outcome is known), the neural network would “learn,” for example, that the presence of cough decreases the likelihood of strep, and the height of fever increases this likelihood.

The appealing feature of neural networks—and what separates this technique from other methods of discovering relationships among data, like logistic regression—is the ability of the system to learn over time. A neural network changes its behavior based on previous patterns. In a domain where the relationship between findings and diseases might change, like

infectious disease surveillance, this changing behavior can be desirable. Another desirable feature of neural networks is the lack of necessity to understand complex relationships between input variables; the network learns these relationships as it changes the links between its nodes. This is the principal difference between neural networks and Bayesian networks. In the latter, one explicitly constructs the network based on one's knowledge of pathophysiology and known probabilities. With neural networks, the links are established as the network is developed, often on the basis of a learning process, without regard to pathophysiologic facts. A disadvantage of neural networks, however, is that unlike the other systems discussed, the "rules" that the network uses do not follow a particular logic and are not explicitly understandable.

Genetic Algorithms

Genetic algorithms represent another nonknowledge-based method for constructing CDSS.²⁸⁻²⁹ Genetic algorithms take their name from an analogy to the molecular rearrangements that take place in chromosomes. Genes rearrange themselves randomly; such rearrangements give rise to variations in an individual, which can affect the individual's ability to pass on genetic material. Over time, the species as a whole incorporates the most adaptive features of the "fittest" individuals. Genetic algorithms take a similar approach. To use a genetic algorithm, the problem to be solved must have many components (e.g., a complex cancer treatment protocol with multiple drugs, radiation therapy, and so on). By selecting components randomly, a population of possible solutions is created. The fittest of these solutions (the one with the best outcome) is selected, and this subpopulation undergoes rearrangement, producing another generation of solutions. By iteratively extracting the best solutions, an optimal solution can be reached. The main challenge in using genetic algorithms is in creating the criteria by which fitness is defined. Since the computing power required to use both genetic algorithms and neural networks is considerable, these techniques have had only limited use in medicine.

Summary

Understanding clinical decision support systems requires a basic understanding of probability and logic. Set theory, familiar to most practitioners who have manipulated collections of literature citations in Medline, provides the basis for understanding probability and other computational methods for reasoning. Probability—in particular, conditional probability—is the principle behind most modern CDSS, but nonprobabilistic heuristic techniques have been used to good effect in the past. Understanding CDSS can be facilitated by considering four basic components of the CDSS

process: input, reasoning engine, knowledge base, and output. Input is often constrained by controlled vocabularies or limitations in temporal expression of clinical features. Reasoning engines take on different designs, but their operation is usually transparent to the user of a CDSS. Knowledge bases contain data from which the reasoning engine takes rules, probabilities, and other constructs required to convert the input into output. Output can take many forms, including a differential diagnosis list or simply a probability of a particular diagnosis. Nonknowledge-based systems use techniques of machine learning to generate methods of turning input into meaningful output, regardless of an explicit representation of expert knowledge.

References

1. Behrman RE, Kliegman RM, Nelson WE, eds. *Nelson textbook of pediatrics* 15th ed. Philadelphia: W.B. Saunders Company; 1996.
2. Shiomi S, Kuroki T, Jomura H, et al. Diagnosis of chronic liver disease from liver scintiscans by fuzzy reasoning. *J Nucl Med* 1995;36:593–598.
3. Suryanarayanan S, Reddy NP, Canilang EP. A fuzzy logic diagnosis system for classification of pharyngeal dysphagia. *Int J Biomed Comput* 1995;38:207–215.
4. Shortliffe EH. *Computer-based medical consultations: MYCIN*. New York: Elsevier; 1976.
5. Kahn MG, Steib SA, Fraser VJ, Dunagan WC. An expert system for culture-based infection control surveillance. *Proc Annu Symp Comput Appl Med Care* 1993;171–175.
6. Emori TG, Culver DH, Horan TC, et al. National nosocomial infections surveillance system (NNIS): description of surveillance methodology. *Am J Infect Control* 1991;19:19–35.
7. Bayes T. An essay towards solving a problem in the doctrine of chances. 1763. *MD Comput* 1991;8:157–171.
8. Warner HR, Toronto AF, Veasey LG, Stephenson R. A mathematical approach to medical diagnosis. Application to congenital heart disease. *JAMA* 1961;177:177–183.
9. Warner HR Jr. Iliad: moving medical decision-making into new frontiers. *Methods Inf Med* 1989;28:370–372.
10. Shortliffe EH, Buchanan BG. A model of inexact reasoning in medicine. *Math Biosci* 1975;23:351–379.
11. Miller R, Masarie FE, Myers JD. Quick medical reference (QMR) for diagnostic assistance. *MD Comput* 1986;3:34–48.
12. Barnett GO, Cimino JJ, Hupp JA, Hoffer EP. DXplain. An evolving diagnostic decision-support system. *JAMA* 1987;258:67–74.
13. Tversky A, Kahneman D. Judgment under uncertainty: heuristics and biases. *Science* 1974;188:1124–1131.
14. Dawes RM, Faust D, Meehl PE. Clinical versus actuarial judgment. *Science* 1989;243:1668–1674.
15. Gigerenzer G, Hoffrage U. How to improve Bayesian reasoning without instruction: frequency formats. *Psychol Rev* 1995;102:684–704.
16. Forsythe DE, Buchanan BG, Osheroff JA, Miller RA. Expanding the concept of medical information: an observational study of physicians' information needs. *Comput Biomed Res* 1992;25:181–200.

17. Aliferis CF, Cooper GF, Miller RA, Buchanan BG, Bankowitz R, Giuse N. A temporal analysis of QMR. *J Am Med Inform Assoc* 1996;3:79-91.
18. Kahn CE Jr, Roberts LM, Wang K, Jenks D, Haddawy P. Preliminary investigation of a Bayesian network for mammographic diagnosis of breast cancer. *Proc Annu Symp Comput Appl Med Care* 1995:208-212.
19. Miller PL, Frawley SJ, Sayward FG, Yasnoff WA, Duncan L, Fleming DW. IMM/Serve: a rule-based program for childhood immunization. *Proc AMIA Annu Fall Symp* 1996:184-188.
20. Porter JF, Kingsland LC 3rd, Lindberg DA, et al. The AI/RHEUM knowledge-based computer consultant system in rheumatology. Performance in the diagnosis of 59 connective tissue disease patients from Japan. *Arthritis Rheum* 1988;31:219-226.
21. Tu SW, Eriksson H, Gennari JH, Shahar Y, Musen MA. Ontology-based configuration of problem-solving methods and generation of knowledge-acquisition tools: application of PROTEGE-II to protocol-based decision support. *Artif Intell Med* 1995;7:257-289.
22. Bar-Hillel M. The base-rate fallacy in probability judgments. *Acta Psychol* 1980;44:211-233.
23. Musen MA, van der Lei J. Knowledge engineering for clinical consultation programs: modeling the application area. *Methods Inf Med* 1989;28:28-35.
24. Mann NH 3d, Brown MD. Artificial intelligence in the diagnosis of low back pain. *Orthop Clin North Am* 1991;22:303-314.
25. Wu Y, Giger ML, Doi K, Vyborny CJ, Schmidt RA, Metz CE. Artificial neural networks in mammography: application to decision making in the diagnosis of breast cancer. *Radiology* 1993;187:81-87.
26. Astion ML, Wener MH, Thomas RG, Hunder GG, Bloch DA. Application of neural networks to the classification of giant cell arteritis. *Arthritis Rheum* 1994;37:760-770.
27. Baxt WG. Use of an artificial neural network for the diagnosis of acute myocardial infarction. *Ann Intern Med* 1991;115:843-848.
28. Levin M. Use of genetic algorithms to solve biomedical problems. *MD Comput* 1995;12:193-199.
29. Grzymala-Busse JW, Woolery LK. Improving prediction of preterm birth using a new classification scheme and rule induction. *Proc Annu Symp Comput Appl Med Care* 1994:730-734.

3

Data Mining and Clinical Decision Support Systems

J. MICHAEL HARDIN and DAVID C. CHHIENG

Introduction

Data mining is a process of pattern and relationship discovery within large sets of data. The context encompasses several fields, including pattern recognition, statistics, computer science, and database management. Thus the definition of data mining largely depends on the point of view of the writer giving the definitions. For example, from the perspective of pattern recognition, data mining is defined as the process of identifying valid, novel, and easily understood patterns within the data set.¹

In still broader terms, the main goal of data mining is to convert data into meaningful information. More specifically, one major primary goal of data mining is to discover new patterns for the users. The discovery of new patterns can serve two purposes: description and prediction. The former focuses on finding patterns and presenting them to users in an interpretable and understandable form. Prediction involves identifying variables or fields in the database and using them to predict future values or behavior of some entities.

Data mining is well suited to provide decision support in the healthcare setting. Healthcare organizations face increasing pressures to improve the quality of care while reducing costs. Because of the large volume of data generated in healthcare settings, it is not surprising that healthcare organizations have been interested in data mining to enhance physician practices, disease management, and resource utilization.

Example 3.1

One early application of data mining to health care was done in the early 1990s by United HealthCare Corporation. United HealthCare Corporation was a managed-care company, and developed its first data mining system, Quality Screening and Management (QSM), to analyze treatment records from its members.² QSM examined 15 measures for studying patients with chronic illness and compared the care received by its members to that recommended by national standards and guidelines. Results of the analyses

were then used to identify appropriate quality management improvement strategies, and to monitor the effectiveness of such actions. Although not providing direct support for decision making at the point of care, these data could be used to improve the way clinical guidelines are used.

Data Mining and Statistical Pattern Recognition

Pattern recognition is a field within the area of data mining. It is the science that seeks analytical models with the ability to describe or classify data/measurements. The objective is to infer from a collection of data/measurements mechanisms to facilitate decision-making processes.^{3,4} With time, pattern recognition methodologies have evolved into an interdisciplinary field that covers multiple areas, including statistics, engineering, computer science, and artificial intelligence. Because of cross-disciplinary interest and participation, it is not surprising that pattern recognition is comprised of a variety of approaches. One approach to pattern recognition is called statistical pattern recognition.

Statistical pattern recognition implies the use of a statistical approach to the modeling of measurements or data.⁵ Briefly, each pattern is represented by a set of features or variables related to an object. The goal is to select features that enable the objects to be classified into one or more groups or classes.

Data Mining and Clinical Decision Support Systems

With the advent of computing power and medical technology, large data sets as well as diverse and elaborate methods for data classification have been developed and studied. As a result, data mining has attracted considerable attention during the past several decades, and has found its way into a large number of applications that have included both data mining and clinical decision support systems. Decision support systems refer to a class of computer-based systems that aids the process of decision making.⁶ Table 3.1 lists some examples of decision support systems that utilize data mining tools in healthcare settings.

A typical decision support system consists of five components: the data management, the model management, the knowledge engine, the user interface, and the user(s).⁷ One of the major differences between decision support systems employing data mining tools and those that employ rule-based expert systems rests in the knowledge engine. In the decision support systems that utilize rule-based expert systems, the inference engine must be supplied with the facts and the rules associated with them that, as described in Chapter 2, are often expressed in sets of “if-then” rules. In this sense, the decision support system requires a vast amount of a priori

TABLE 3.1. Examples of clinical decision support systems and data mining tools that utilize statistical pattern recognition.

System (reference)	Description
Medical imaging recognition and interpretation system	
Computer-aided diagnosis of melanoma ²³	Analysis of digitized images of skin lesions to diagnose melanoma
Computer-aided diagnosis of breast cancer ²¹	Differentiation between benign and malignant breast nodules, based on multiple ultrasonographic features
Monitoring tumor response to chemotherapy ³⁰	Computer-assisted texture analysis of ultrasound images aids monitoring of tumor response to chemotherapy
Diagnosis of neuromuscular disorder ³¹	Classification of electromyographic (EMG) signals, based on the shapes and firing rates of motor unit action potentials (MUAPs)
Discrimination of neoplastic and non-neoplastic brain lesions ²⁷	Predicting the presence of brain neoplasm with magnetic resonance spectroscopy
Gene and protein expression analysis	
Molecular profiling of breast cancer ²⁵	Identification of breast cancer subtypes distinguished by pervasive differences in their gene expression patterns
Screening for prostate cancer ³²	Early detection of prostate cancer based on serum protein patterns detected by surface enhanced laser description ionization time-of-flight mass spectrometry (SELDI-TOF MS)
Educational system	
Mining biomedical literature ³³	Automated system to mine MEDLINE for references to genes and proteins and to assess the relevance of each reference assignment
Laboratory system	
ISPAHAN ³⁴	Classification of immature and mature white blood cells (neutrophils series) using morphometrical parameters
Histologic diagnosis of Alzheimer's disease ³⁵	Analysis of digital images of tissue sections to identify and quantify senile plaques for diagnosing and evaluating the severity of Alzheimer's disease
Diagnosis of inherited metabolic diseases in newborns ³⁶	Identification of novel patterns in high-dimensional metabolic data for the construction of classification system to aid the diagnosis of inherited metabolic diseases
Acute care system	
Identification of hospitals with potential quality problems ³⁷	Using logistic regression models to compare hospital profiles based on risk-adjusted death with 30 days of noncardiac surgery
Prediction of disposition for children with bronchiolitis ²²	Neural network system to predict the disposition in children presenting to the emergency room with bronchiolitis
Estimating the outcome of hospitalized cancer patients ²⁸	Predicting the risk of in-hospital mortality in cancer patients with nonterminal disease
Miscellaneous	
Flat foot functional evaluation ³⁸	Gait analysis to diagnosis "flat foot" and to monitor recovery after surgical treatment

knowledge on the part of the decision maker in order to provide the right answers to well formed questions. On the contrary, the decision support systems employing data mining tools do not require a priori knowledge on the part of the decision maker. Instead, the system is designed to find new and unsuspected patterns and relationships in a given set of data; the system then applies this newly discovered knowledge to a new set of data. This is most useful when a priori knowledge is limited or nonexistent.

Many successful clinical decision support systems using rule-based expert systems have been developed for very specialized areas in health care.⁸⁻¹⁴ One early example of a rule-based expert system is MYCIN, which used its rules to identify micro-organisms that caused bacteremia and meningitis.¹⁴ However, such systems can be challenging to maintain due to the fact that they often contain several thousand rules or more. In addition, these “if-then” rule systems have difficulty dealing with uncertainty. Bayesian systems (see Chapter 2) are one way of addressing uncertainty. Statistical pattern recognition approaches are another.

Supervised Versus Unsupervised Learning

Data mining and predictive modeling can be understood as learning from data. In this context, data mining comes in two categories: supervised learning and unsupervised learning.

Supervised Learning

Supervised learning, also called directed data mining, assumes that the user knows ahead of time what the classes are and that there are examples of each class available. (Figure 3.1A) This knowledge is transferred to the system through a process called training. The data set used in this process is called the training sample. The training sample is composed of dependent or target variables, and independent variables or input. The system is adjusted based on the training sample and the error signal (the difference between the desired response and the actual response of the system). In other words, a supervised learning system can be viewed as an operation that attempts to reduce the discrepancy between the expected and observed values as the training process progresses. With enough examples in the training data, the discrepancy will be minimized and the pattern recognition will be more accurate.

The goal of this approach is to establish a relationship or predictive model between the dependent and independent variables. Predictive modeling falls into the category of supervised learning because one variable is designated as the target that will be explained as a function of other variables. Predictive models are often built to predict the future values or behavior of an object or entity. The nature of the target/dependent variable

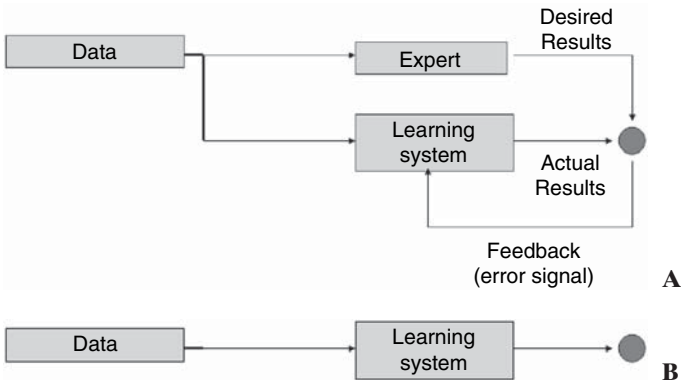


FIGURE 3.1. A, Supervised learning. B, Unsupervised learning.

determines the type of model: a model is called a classification model if the target variable is discrete; and a regression model if the target variable is continuous.

Example 3.2

Goldman et al. described the construction of a clinical decision support system to predict the presence of myocardial infraction in a cohort of 4,770 patients presenting with acute chest pain at two university hospitals and four community hospitals.¹⁵ Based on the patient’s symptoms and signs, the clinical decision support system had similar sensitivity (88.0% versus 87.8%) but a significantly higher specificity (74% versus 71%) in predicting the absence of myocardial infarction when compared to physicians’ decisions if the patients were required to be admitted to the coronary care unit. If the decision to admit was based solely on the decision support system, the admission of patients without infarction to the coronary care unit would have been reduced by 11.5% without adversely affecting patient outcomes or quality of care.

A Priori Probability

In supervised learning, the frequency distribution, or a priori probability, of the classes of a certain training set (or a sample taken from the general population) may be quite different from that of the general population to which the classifier is intended to be applied. In other words, the training set/sample may not represent the general population. For example, a par-

ticular training set may consist of 50% of the subjects with disease and 50% without the disease. In this case, a priori probabilities of the two classes in the training set are 0.5 for each class. However, the actual a priori probability or the actual prevalence of disease may be very different (less than or greater than 0.5) from that of the training set. In some instances, the actual a priori probability of the general population may be unknown to the researchers. This may have a negative effect on the performance of the classifier when applied to a real world data set. Therefore, it is necessary to adjust the output of a classifier with respect to the new condition to ensure the optimal performance of the classifier.¹⁶

Unsupervised Learning

In unsupervised or undirected learning, the system is presented with a set of data but no information is available as to how to group the data into more meaningful classes (Figure 3.1B). Based on perceived similarities that the learning system detects within the data set, the system develops classes or clusters until a set of definable patterns begins to emerge. There are no target variables; all variables are treated the same way without the distinction between dependent and independent variables.

Example 3.3

Avanzolini et al. analyzed 13 commonly monitored physiological variables in a group of 200 patients in the six-hour period immediately following cardiac surgery in an attempt to identify patients who were at risk for developing postoperative complications.¹⁷ Using an unsupervised learning (clustering) method, the investigators showed the existence of two well defined categories of patients: those with low risk of developing postoperative complications and those with high risk.

Classifiers for Supervised Learning

In supervised learning, classification refers to the mapping of data items into one of the predefined classes. In the development of data mining tools and clinical decision support systems that use statistical approaches like those described here, one of the critical tasks is to create a classification model, known as a classifier, which will predict the class of some entities or patterns based on the values of the input attributes. Choosing the right classifier is a critical step in the pattern recognition process. A variety of techniques have been used to obtain good classifiers. Some of the more widely used and well known techniques that are used in data mining include decision trees, logistic regression, neural networks, and nearest neighbor approach.

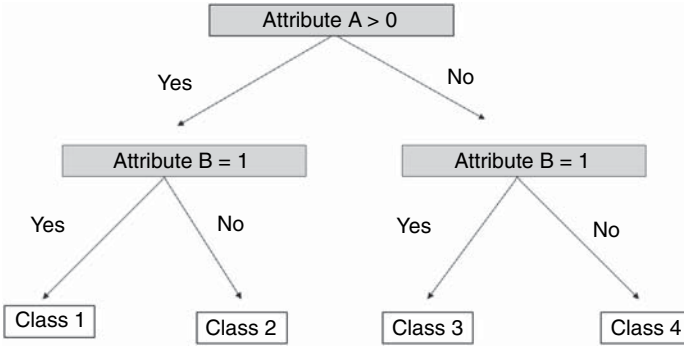


FIGURE 3.2. A simple decision tree with the tests on attributes A and B.

Decision Trees

The use of decision trees is perhaps the easiest to understand and the most widely used method that falls into the category of supervised learning. Figure 3.2 is the graphical representation of a simple decision tree using two attributes. A typical decision tree system adopts a top-down strategy in searching for a solution. It consists of nodes where predictor attributes are tested. At each node, the algorithm examines all attributes and all values of each attribute with respect to determining the attribute and a value of the attribute that will “best” separate the data into more homogeneous subgroups with respect to the target variable. In other words, each node is a classification question and the branches of the tree are partitions of the data set into different classes. This process repeats itself in a recursive, iterative manner until no further separation of the data is feasible or a single classification can be applied to each member of the derived subgroups. Therefore, the terminal nodes at the end of the branches of the decision tree represent the different classes.

Example 3.4

An example of a clinical decision support system using decision trees can be found in a study by Gerald et al.¹⁸ The authors developed a decision tree that assisted health workers in predicting which contacts of tuberculosis patients were most likely to have positive tuberculin skin tests. The model was developed based on 292 consecutive cases and close to 3,000 contacts and subsequently tested prospectively on 366 new cases and 3,162 contacts. Testing showed that the decision tree model had a sensitivity of 94%, a specificity of 28%, and a false negative rate of 7%. The authors concluded that the use of decision trees would decrease the number of contacts investigated by 25% while maintaining a false negative rate that was close to

that of the presumed background rate of latent tuberculosis infection in the region.

Logistic Regression

Logistic regression is used to model data in which the target or dependent variable is binary, i.e., the dependent variable can take the value 1 with a probability of success p , or the value 0 with the probability of failure $1 - p$. The main objective is to develop a regression type model relating the binary variable to the independent variables. As such it is a form of supervised learning. It can also be used to examine the variation in the dependent variable that can be explained by the independent variables, to rank the independent variables based on their relative importance in predicting the target variable, and to determine the interaction effects among independent variables. Rather than predicting the values of the dependent variable, logistic regression estimates the probability that a dependent variable will have a given value. For example, instead of predicting whether a patient is suffering from a certain disease, logistic regression tries to estimate the probability of the patient having the disease. If the estimated probability is greater than 0.5, then there is a higher probability of the patient having the disease than not having the disease. The function relating the probabilities to the independent variables is not a linear function and is represented by the following equation:

$$p(y) = 1/\{1 + e^{(-a-bx)}\}$$

where $p(y)$ is the probability that y , the dependent variable, occurs based on x , the value of an attribute/independent variable, a is the constant, and b is the coefficient of the independent variable. Figure 3.3 shows a graphical representation of the logistic regression model which fits the relationship between the value of the independent variable, x and the probability of dependent variable, y occurring with a special S-shaped curve that is mathematically constrained to remain within the range of 0.0 to 1.0 on the Y axis.

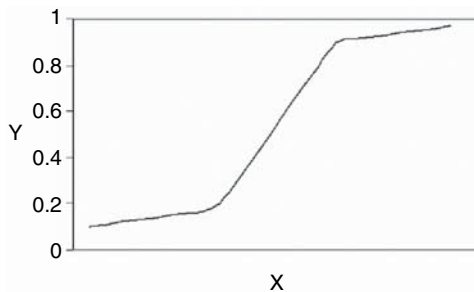


FIGURE 3.3. Logistic regression model.

Example 3.5

The following is an example that applies logistic regression to decision making. In the earliest stage of the epidemic of severe acute respiratory syndrome (SARS) when reliable rapid confirmatory tests were lacking, a group of researchers from Taiwan attempted to establish a scoring system to improve the diagnostic accuracy of SARS.¹⁹ The scoring system was developed based on the clinical and laboratory findings of 175 suspected cases using a multivariate, stepwise logistic regression model. The authors then applied the scoring system to 232 patients and were able to achieve a sensitivity and specificity of 100% and 93%, respectively, in diagnosing SARS.

Example 3.6

In another study, the authors applied texture analysis to images of breast tissue generated by magnetic resonance imaging (MRI) for differentiating between benign and malignant lesions.²⁰ Using logistic regression analysis, a diagnostic accuracy of 0.8 +/- 0.07 was obtained with a model requiring only three parameters.

Neural Networks

The original development of the neural network programs was inspired by the way the brain recognizes patterns. A neural network is composed of a large number of processors known as neurons (after the brain cells that perform a similar function) that have a small amount of local memory and are connected unidirectionally (Figure 3.4). Each neuron can have more than one input and operates only on the inputs it receives via the connections. Like some of the data mining tools, neural networks can be supervised or unsupervised. In supervised neural networks, examples in the form of the training data are provided to the network one at a time. For each example, the network generates an output that is compared with the actual

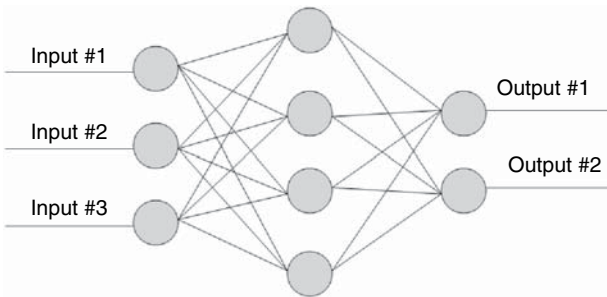


FIGURE 3.4. Neural network.

value as a form of feedback. Once the output of the neural network is the same as the actual value, no further training is required. If the output differs from the actual value, the network adjusts those parameters that contributed to the incorrect output. Once adjustment is made, another example is presented to the network and the whole process is repeated. The process terminates when all parameters are stabilized. The size and representativeness of the training data is obviously very important, since a neural network could work fine on the training set, but not generalize to a broader sample.

Example 3.7

One example of a neural network is the computer-aided diagnosis of solid breast nodules. In one study, ultrasonographic features were extracted from 300 benign and 284 malignant biopsy-confirmed breast nodules.²¹ The neural network was trained with a randomly selected data set consisting of half of the breast nodule ultrasonographic images. Using the trained neural network, surgery could be avoided in over half of the patients with benign nodules with a sensitivity of 99%.

Example 3.8

In another example, a neural network was used to detect the disposition in children presenting to the emergency room with bronchiolitis (inflammation of small airways).²² The neural network correctly predicted the disposition in 81% of test cases.

Nearest Neighbor Classifier

When a system uses the nearest neighbor (NN) classification, each attribute is assigned a dimension to form a multidimensional space. A training set of objects, whose classes are known, are analyzed for each attribute; each object is then plotted within the multidimensional space based on the values of all attributes. New objects, whose classes are yet to be determined, are then classified according to a simple rule; each new object is analyzed for the same set of attributes and is then plotted within the multidimensional space based on the value of each attribute. The new object is assigned to the same class of its closest neighbor based on appropriate metric/measurements. In other words, the NN rule assumes that observations which are the closest together (based on some form of measurement) belong to the same category (Figure 3.5). The NN rule is often used in situations where the user has no knowledge of the distribution of the categories.

One extension of this approach is the k-nearest neighbor approach (k-NN). Instead of comparing to a single nearest prototype, one can take into account k-neighboring points when classifying a data point, if the number of preclassified points is large. For each new pattern, the class is assigned by finding the most prominent class among the k-nearest data points in the

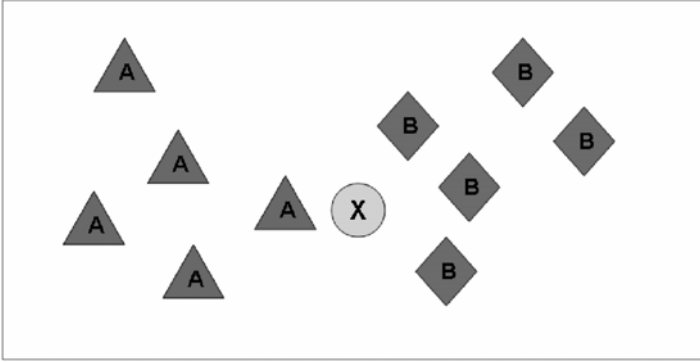


FIGURE 3.5. Nearest neighbor (NN) classifier. There are two classes: A (triangles) and B (diamonds). The circle represents the unknown sample, X. For the NN rule, the nearest neighbor of X comes from class A, so it would be labeled class A. Using the k-NN rule with $k = 4$, three of the nearest neighbors of sample X come from class B, so it would be labeled as B.

training set. (Figure 3.5) This approach works very well in cases where a class does not form a single coherent group but is a collection of more than one separate group.

Example 3.9

By applying the k-NN classifier, Burrone et al. developed a decision support system to assist clinicians with distinguishing early melanoma from benign skin lesions, based on the analysis of digitized images obtained by epiluminescence microscopy.²³ Digital images of 201 melanomas and 449 benign nevi were included in the study and were separated into two groups, a learning set and a test set. A k-NN pattern recognition classifier was constructed using all available image features and trained for a sensitivity of 98% with the learning set. Using an independent test set of images, a mean specificity of 79% was achieved with a sensitivity of 98%. The authors concluded that this approach might improve early diagnosis of melanoma and reduce unnecessary surgery.

Evaluation of Classifiers

ROC Graphs

In statistical pattern recognition, the goal is to map entities to classes. Therefore, the ultimate question is: which classifiers are more accurate in performing this classification task? Suppose one wanted to identify which classifiers would be best to determine whether a patient has cancer or not,

based on the results of certain laboratory tests. Given a classifier and an instance, there are four possible outcomes. If the patient has cancer and is diagnosed with cancer, based on the classifier, it is considered a true positive; if the patient is declared healthy by the classifier, but really has cancer, it is considered a false negative. If the patient has no cancer and is declared healthy, it is considered a true negative; if he is diagnosed as having cancer when he is really healthy, it is considered a false positive.

We can plot the true positive rate on the Y axis and the false positive rate on the X axis; a receiver operating characteristic (ROC) graph results (Figure 3.6). The true positive rate (also known as sensitivity) is obtained by dividing the number of true positives by the sum of true positives and false negatives. The false positive rate is obtained by dividing the number of false positives divided by the sum of true negatives and false positives; the false positive rate can also be expressed as “1 minus specificity,” where specificity is equal to true negatives divided by the sum of true negatives and false positives. The ROC graph is a two-dimensional graph that depicts the trade-offs between benefits (detecting cancer correctly, or true positive) and costs (false alarm or false positive). Each classifier generates a pair of true positive and false positive rates, which corresponds to a point on the ROC graph. The point (0, 1) represents perfect classification, i.e., 100% true positive rate and 0% false positive rate. One classifier is considered superior to another if it has a higher true positive rate and a lower false positive rate, corresponding to a more “northwest” location relative to the other on the ROC graph. In general, the false alarm rates go up as one attempts to increase the true positive rate. Classifiers with points on the southwest corner of an ROC graph are more “conservative” since they make positive predictions only with strong evidence; therefore there is a low true positive rate, but also few false positive errors. On the other hand, classifiers on the

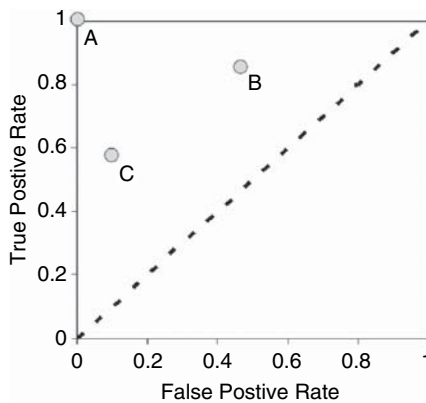


FIGURE 3.6. Receiver operating characteristic (ROC) curve. Point A represents perfect performance. The performance of C is more conservative than B.

northeast corner of an ROC graph are more “liberal” since they make positive prediction with weak evidence; therefore they have high true positive rates, but also high false positive rates.

Some classifiers, such as neural networks, yield a numeric value which can be in the form of a numeric score or probability that represents the likelihood an object belongs to a certain class. These classifiers can be converted into discrete, binary (yes versus no) classifiers by setting a threshold, i.e., if the output score is above the threshold, the classifier produces a “Yes, else a No”. By choosing a different threshold, a different point in the ROC graph is produced. As a result, varying the thresholds will produce a curve in the ROC graph for a particular classifier. Given an ROC curve, one can select the threshold corresponding to a particular point on the ROC that produces the desired binary classifier with the best true positive rate (correctly diagnosed cancer) within the constraints of an acceptable false positive rate (false alarm). This is chosen based on the relative costs of the two types of errors: missing a diagnosis of cancer (type I error) versus creating a false alarm (type II error).

The area under the ROC curve (AUC) provides a single statistic (the C-Statistic) for comparing classifiers. It measures the accuracy of the classifiers. Consider the situation in which a classifier attempts to separate patients into two groups; those with disease and those without. One can randomly pick a patient from the disease group and one from the non-disease group and apply the classifier on both. The area under the curve represents the percentage of randomly drawn pairs where the classifier correctly classifies the two patients in the random pair. The value of AUC ranges from 0.5 to 1. A classifier with an AUC of 0.5 would be a poor classifier, roughly equivalent to flipping a coin to decide the class membership. A classifier with an AUC close to 1 results in better classification of entities to classes. For example, in Example 3.6, the resulting trained neural network model yielded a normalized area under the ROC curve of 0.95.

Computing the AUC is complex and beyond the scope of this chapter. Briefly, there are two commonly used methods. One method is based on the construction of trapezoids under the curve as an approximation of the area. The other method employs a maximum likelihood estimator to fit a smooth curve to the data points. Both methods are available as computer programs and give an estimate of area and standard error that can be used to compare different classifiers.

Kolmogorov-Smirnov Test

While the AUC provides a way of distinguishing groups overall, there are other statistical tests used to provide a more refined comparison of groups or subgroups. The Kolmogorov-Smirnov test, or KS test, is used to determine whether the distributions of two samples differ from each other or

whether the distribution of a sample differs from that of the general population. The KS test provides what is called the D-statistic for comparison of classifiers.²⁴

Unsupervised Learning

Cluster Analysis

Unsupervised classification refers to situations where the goal is to classify a diverse collection of unlabeled data into different groups based on different features in a data set. Unsupervised classification, also known as cluster analysis or clustering, is a general term to describe methodologies that are designed to find natural groupings or clusters based on measured or perceived similarities among the items in the clusters using a multidimensional data set (Figure 3.7). There is no need to identify the groupings desired or the features that should be used to classify the data set. In addition, clustering offers a generalized description of each cluster, resulting in better understanding of the data set's characteristics and providing a starting point for exploring further relationships.

Clustering techniques are very useful in data mining because of the speed, reliability, and consistency with which they can organize a large amount of data into distinct groupings. Despite the availability of a vast collection of clustering algorithms in the literature, they are based on two popular approaches: hierarchical clustering and nonhierarchical clustering. The former, which is the most frequently used technique, organizes data in a nested sequence of groups that can be displayed in a tree-like structure, or dendrogram.

There are several problems that are associated with clustering. One problem is that data can be grouped into clusters with different shapes and sizes. Another problem is the resolution or granularity, i.e., fine versus

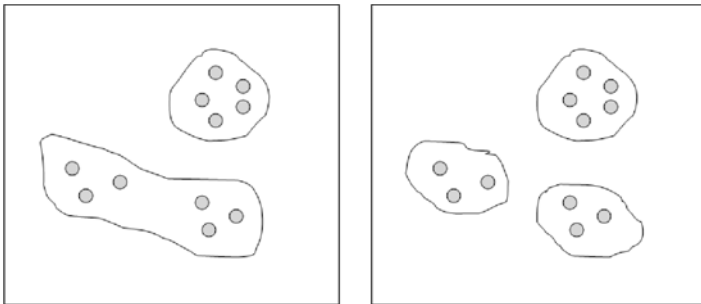


FIGURE 3.7. Cluster analysis. Two clusters of data (left); three clusters (right) using the same set of data.

coarse, with which the data are viewed. This problem is most obvious when one tries to delineate a region containing a high density of patterns compared to the background. Therefore, some authors define a cluster as one that consists of a relatively high density of points separated from other clusters by a relatively low density of points, whereas some define clusters containing samples that share more similarities to each other than to samples of different clusters. As a result, the selection of an appropriate measure of similarity to define clusters is a major challenge in cluster analysis.

Gene Expression Data Analysis

One of the applications of cluster analysis in medicine is the analysis of gene expression. With the completion of the human genome project, which identified more than 30,000 gene sequences, researchers are now able to examine the expression of several thousand genes from blood, body fluids, and tissue samples at the same time, in an attempt to identify gene subsets that are associated with various disease statistics. Since information is obtained from hundreds and thousands of gene sequences, an astronomical body of data is generated. Common research questions often fall under the following categories: class discovery, class prediction, and gene identification. Class prediction refers to the classification of samples based on certain behaviors or properties such as response to therapy, whereas gene identification involves the discovery of genes that are differentially expressed among different disease groups.

Class discovery refers to the discovery of previously unknown categories or subtypes based on some similarity measure calculated from the gene expression data. Cluster analysis is often the method of choice in accomplishing this task, because samples are clustered into groups based on the similarity of their gene expressions without utilizing any knowledge of any predefined classification schemes such as known histological tumor classification.

Example 3.10

In the future, it is likely that genomic data will be incorporated into clinical decision support systems to refine both diagnosis and therapy. The following is an example that used clustering to explore breast cancer classification using genomic data. In this study, Perou et al. evaluated the pattern of gene expression of 8,102 human genes in 65 breast cancers obtained from 42 patients.²⁵ Using hierarchical cluster analysis, the authors were able to classify 65 breast cancer samples into three distinct subtypes. One subtype was cancers that overexpressed the oncogene *erbB-2*. The remaining two subtypes were unknown prior to this study; they were estrogen receptor-positive luminal-like cancers and basaloid cancers. Subsequent survival analyses on a group of patients with locally advanced breast cancer

showed significantly different outcomes for the patients belonging to different subtypes; patients with basaloid cancers had a poor survival rate.²⁶ In the same study by Perou et al, the samples contained 20 primary tumors that were biopsied twice, before and after the completion of chemotherapy. Using clustering, the authors demonstrated that gene expression patterns were similar among samples from the same patients taken at different time points but not between samples taken from different patients.

Other Techniques

The goal of any tool that is used for pattern recognition is to arrive at an optimal solution within a given set of complex constraints. The development of sophisticated computer-based computation techniques has enabled analysts to attain better solutions than previous techniques. As improved techniques are developed to handle increasingly complex problems, there is a corresponding need for more innovative methods for arriving at optimal solutions. Genetic algorithms and biologic computing are two examples of innovative techniques that have gained increasing acceptance and application in the field of pattern recognition and data mining.

Genetic Algorithms

The fundamental concept of genetic algorithms has its roots in Darwin's evolutionary theories of natural selection and adaptation. According to Darwin, organisms that come up with successful solutions to best support them and protect themselves from harm survive, whereas those organisms that fail to adapt to their environment become extinct. Based on the same idea of "survival of the fittest," a genetic algorithm initially tries to solve a given problem with random solutions. These solutions are often referred to as the genomes, or a collection of genes. The gene represents the smallest unit of information for the construction of possible solutions. The next step is to evaluate or quantify the fitness of all the available genomes or solutions based on a fitness function. The latter returns a value of goodness or fitness so that a particular genome or solution may be ranked against all other genomes or solutions. Those solutions with better fit are ranked higher among others and are allowed to "breed." Once the initial evaluation is completed, the genetic algorithms examine new solutions by letting all the current solutions "evolve" through mutual exchange of "genetic materials" among solutions to improve the genomes and/or mutation (i.e., randomly changing the genetic materials) to "create" new solutions. The new solutions are then evaluated using the same fitness functions to determine which solutions are good and which are not and need to be eliminated. Thus the process repeats itself until an "optimal" solution is attained.

There are many benefits of genetic algorithms. One major advantage is that a genetic algorithm almost always guarantees finding some reasonable solution to problems, particularly those that we have no idea how to solve. Further, the final solution is often superior to the initial collection of possible solutions. Another benefit is that genetic algorithms tend to arrive at a solution much faster than other optimization techniques. Also, the strength of the genetic algorithm does not depend upon complex algorithms but rather on relatively simple concepts. Despite the power of genetic algorithms, however, some parameters, such as the size of the solution population, the rate of mutation and crossover, and the selection methods and criteria, can significantly affect their performance. For example, if the solution population size is too small, the genetic algorithm may have exhausted all the available solutions before the process can identify an optimal solution. If the rate of genetic mutation is too high, the process may be changing too fast for the selection to ever bring about convergence, resulting in the failure of generating an optimal solution.

Example 3.11

Genetic algorithms have been used to construct clinical decision support systems. In a study by Zellner et al., the authors evaluated the performance of a logistic regression model in diagnosing brain tumors with magnetic resonance spectroscopy using the genetic algorithms approach.²⁷ The genetic algorithm approach was superior to the conventional approach in 14 out of 18 trials, and the genetic algorithm had fewer false negatives and false positives. In addition, the authors also pointed out that the genetic algorithm approach was less costly.

Example 3.12

Genetic algorithms have also been used as a data mining technique in healthcare operations. One study investigated whether genetic algorithms could be used to predict the risk of in-hospital mortality of cancer patients.²⁸ A total of 201 cancer patients, over a two-year period of time, was retrospectively evaluated. Compared to other methods, such as multivariate logistic regression, neural networks, and recursive partitioning analysis, genetic algorithms selected the least number of explanatory variables with a comparable proportion of the cases explained (79%). The authors concluded that genetic algorithms reliably predicted in-hospital mortality of cancer patients and were as efficient as the other data mining techniques examined.

Biological Computing

Biological computing is another new discipline that has found its way into data mining applications. It cuts across two well established fields: computer

science and biology. While the genetic algorithm approach uses the analogy of natural selection to develop computer algorithms, the idea of biological computing actually involves the use of living organisms or their components, e.g., DNA strands, to perform computing operations. The benefits include the ability to hold enormous amounts of information, the capability of massive parallel processing, self-assembly, self-healing, self-adaptation, and energy efficiency. As of now, a biological computer can only perform rudimentary functions and it has no practical applications, but its potential continues to emerge. For example, some scientists have been working on the development of tiny DNA computers that circulate in a person's body to monitor his/her well-being and release the right drugs to repair damaged tissue or fight off infections and cancers.²⁹

Conclusions

Data mining refers to the process of pattern and relationship discovery within large data sets. It holds promise in many areas of health care and medical research, with applications ranging from medical diagnosis to quality assurance. The power of data mining lies in its ability to allow users to consider data from a variety of perspectives in order to discover apparent or hidden patterns. There are two main divisions of classification: supervised learning or training, and unsupervised learning. Supervised training requires training samples to be labeled with a known category or outcome to be applied to the classifier. There are many classifiers available and their performance can be assessed using an ROC curve. Unsupervised learning, also known as clustering, refers to methodologies that are designed to find natural groupings or clusters without the benefit of a training set. The goal is to discover hidden or new relationships within the data set. One application of clustering is the analysis of gene expression data. Genetic algorithms and biological computing are two newer disciplines that have found their way into data mining applications and clinical decision support systems.

References

1. Fayyad UM, Piatetsky-Shapiro G, Smyth P. Knowledge discovery and data mining: towards a unifying framework. Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining, pp. 82–88. Portland, Oregon, August 1996. AAAI Press. Available from: <http://ww-aig.jpl.nasa.gov/kdd96>. Accessed July 17, 2006.
2. Leatherman S, Peterson E, Heinen L, Quam L. Quality screening and management using claims data in a managed care setting. QRB Qual Rev Bull 1991; 17:349–359.
3. Duda RO, Hart PE, Stork DG. Pattern classification and scene analysis, 2nd ed. New York: John Wiley and Sons; 2000.

4. Fukunaga K. Introduction to statistical pattern recognition, 2nd ed. New York: Academic Press; 1990.
5. Schalkoff RJ. Pattern recognition: statistical, structural and neural approaches. New York: John Wiley and Sons; 1991.
6. Finlay PN. Introducing decision support systems. Cambridge, MA: Blackwell Publishers; 1994.
7. Marakas GM. Decision support systems, 2nd ed. Princeton, NJ: Prentice Hall; 2002.
8. Ambrosiadou BV, Goulis DG, Pappas C. Clinical evaluation of the DIABETES expert system for decision support by multiple regimen insulin dose adjustment. *Comp Methods Programs Biomed* 1996;49:105–115.
9. Marchevsky AM, Coons G. Expert systems as an aid for the pathologist's role of clinical consultant: CANCER-STAGE. *Mod Pathol* 1993;6:265–269.
10. Nguyen AN, Hartwell EA, Milam JD. A rule-based expert system for laboratory diagnosis of hemoglobin disorders. *Arch Pathol Lab Med* 1996;120:817–827.
11. Papaloukas C, Fotiadis DI, Likas A, Stroumbis CS, Michalis LK. Use of a novel rule-based expert system in the detection of changes in the ST segment and the T wave in long duration ECGs. *J Electrocardiol* 2002;35:27–34.
12. Riss PA, Koelbl H, Reinthaller A, Deutinger J. Development and application of simple expert systems in obstetrics and gynecology. *J Perinat Med* 1988;16: 283–287.
13. Sailors RM, East TD. A model-based simulator for testing rule-based decision support systems for mechanical ventilation of ARDS patients. *Proc Ann Symp Comp Appl Med Care* 1994:1007.
14. Shortliffe EH, Davis R, Axline SG, Buchanan BG, Green CC, Cohen SN. Computer-based consultations in clinical therapeutics: explanation and rule acquisition capabilities of the MYCIN system. *Comput Biomed Res* 1975; 8:303–320.
15. Goldman L, Cook EF, Brand DA, et al. A computer protocol to predict myocardial infarction in emergency department patients with chest pain. *N Engl J Med* 1988;318:797–803.
16. Scott AJ, Wild CJ. Fitting logistic models under case-control or choice based sampling. *J Roy Stat Soc B* 1986;48:170–182.
17. Avanzolini G, Barbini P, Gnudi G. Unsupervised learning and discriminant analysis applied to identification of high risk postoperative cardiac patients. *Int J Biomed Comp* 1990;25:207–221.
18. Gerald LB, Tang S, Bruce F, et al. A decision tree for tuberculosis contact investigation [see comment]. *Am J Respir Crit Care Med* 2002;166:1122–1127.
19. Wang TL, Jang TN, Huang CH, et al. Establishing a clinical decision rule of severe acute respiratory syndrome at the emergency department. *Ann Emerg Med* 2004;43:17–22.
20. Gibbs P, Turnbull LW. Textural analysis of contrast-enhanced MR images of the breast. *Magn Reson Med* 2003;50:92–98.
21. Joo S, Yang YS, Moon WK, Kim HC. Computer-aided diagnosis of solid breast nodules: use of an artificial neural network based on multiple sonographic features. *IEEE Transact Med Imaging* 2004;23:1292–1300.
22. Walsh P, Cunningham P, Rothenberg SJ, O'Doherty S, Hoey H, Healy R. An artificial neural network ensemble to predict disposition and length of stay in children presenting with bronchiolitis. *Eur J Emerg Med* 2004;11:259–564.

23. Burroni M, Corona R, Dell'Eva G, et al. Melanoma computer-aided diagnosis: reliability and feasibility study. *Clin Cancer Res* 2004;10:1881–1886.
24. Press WH, Flannery BP, Teukolsky SA, Vetterling WT. Numerical recipes in FORTRAN example book: the art of scientific computing. 2nd Ed New York: Cambridge University Press; 1992.
25. Perou CM, Sorlie T, Eisen MB, et al. Molecular portraits of human breast tumours. *Nature* 2000;406:747–752.
26. Sorlie T, Perou CM, Tibshirani R, et al. Gene expression patterns of breast carcinomas distinguish tumor subclasses with clinical implications. *Proc Natl Acad Sci USA* 2001;98:10869–10874.
27. Zellner BB, Rand SD, Prost R, Krouwer H, Chetty VK. A cost-minimizing diagnostic methodology for discrimination between neoplastic and non-neoplastic brain lesions: utilizing a genetic algorithm. *Acad Radiol* 2004;11:169–177.
28. Bozcuk H, Bilge U, Koyuncu E, Gulkesen H. An application of a genetic algorithm in conjunction with other data mining methods for estimating outcome after hospitalization in cancer patients. *Med Sci Monit* 2004;10:CR246–CR251.
29. Benenson Y, Gil B, Ben-Dor U, Adar R, Shapiro E. An autonomous molecular computer for logical control of gene expression [see comment]. *Nature* 2004; 429:423–429.
30. Huber S, Medl M, Vesely M, Czembirek H, Zuna I, Delorme S. Ultrasonographic tissue characterization in monitoring tumor response to neoadjuvant chemotherapy in locally advanced breast cancer (work in progress). *J Ultrasound Med* 2000;19:677–686.
31. Christodoulou CI, Pattichis CS. Unsupervised pattern recognition for the classification of EMG signals. *IEEE Trans Biomed Eng* 1999;46:169–178.
32. Banez LL, Prasanna P, Sun L, et al. Diagnostic potential of serum proteomic patterns in prostate cancer. *J Urol* 2003;170(2 Pt 1):442–426.
33. Leonard JE, Colombe JB, Levy JL. Finding relevant references to genes and proteins in Medline using a Bayesian approach. *Bioinformatics* 2002;18:1515–1522.
34. Bins M, van Montfort LH, Timmers T, Landeweerd GH, Gelsema ES, Halie MR. Classification of immature and mature cells of the neutrophil series using morphometrical parameters. *Cytometry* 1983;3:435–438.
35. Hibbard LS, McKeel DW Jr. Automated identification and quantitative morphometry of the senile plaques of Alzheimer's disease. *Anal Quant Cytol Histol* 1997;19:123–138.
36. Baumgartner C, Bohm C, Baumgartner D, et al. Supervised machine learning techniques for the classification of metabolic disorders in newborns. *Bioinformatics* 2004;20:2985–2996.
37. Gordon HS, Johnson ML, Wray NP, et al. Mortality after noncardiac surgery: prediction from administrative versus clinical data. *Med Care* 2005;43:159–167.
38. Bertani A, Cappello A, Benedetti MG, Simoncini L, Catani F. Flat foot functional evaluation using pattern recognition of ground reaction data. *Clin Biomech* 1999;14:484–493.

4

Design and Implementation Issues

JEROME H. CARTER

The early 1970s were a time of great optimism for researchers in the field of medical artificial intelligence. The initial successes of systems such as MYCIN,¹ CASNET,² and the Leeds abdominal pain system³ made it reasonable to assume that it was only a matter of time until computers became a standard part of physicians' diagnostic armamentarium. Over the past few years, the emphasis in clinical decision support has shifted from its initial narrow focus on diagnostic expert systems to a much broader range of applications. Increasingly, clinicians have access to alerts, reminders, and patient-specific advice for such common tasks as prescription writing and test ordering.^{4,5,6,7} Despite these gains, CDSS are not yet common in patient care settings.⁸ This chapter will examine the key design and implementation concerns that must be addressed if these systems are to realize their full potential.

What accounts for this lack of use? The 30-year experience described by Engle⁹ provides valuable insight into the problems encountered in the creation and deployment of CDSS. He provides a list of factors divided into critical and noncritical that he feels account for the difficulties in building a useful system and the rejection of diagnostic systems by clinicians. According to Engle, "Factors that play a role but are not critical include inadequate computers and peripheral devices, difficulty some people have working with computers, systems not user-friendly, physicians' high regard for their own capabilities, and fear of computer competition, as well as the limited nature of the programs. In our estimation, the critical impediment to the development of decision programs useful in medicine lies in the impossibility of developing an adequate database and an effective set of decision rules." The findings of Berner et al.¹⁰ help us to understand some of the frustration noted by Engle. In their test of four general diagnostic systems, Berner et al. found that "... the proportion of correct diagnoses ranged from 0.52–0.71 and the mean proportion of relevant diagnoses ranged from 0.19–0.37. . . ." This is hardly the type of performance which encourages use by a busy clinician. While this level of performance may be problematic for the broad-based systems like QMR,¹¹ Iliad,¹² and DXplain,¹³ programs with

more limited domains such as Pathfinder¹⁴ and the Leeds abdominal pain system³ have been noted to perform very well. However, Shortliffe¹⁵ properly notes that systems dedicated to a single problem tend to discourage wide usage because of their limited scope.

Another major design issue is the lack of integration into standard information systems,¹⁵⁻¹⁷ although this is beginning to change. In contrast to the lack of acceptance of expert systems in the field of medicine, other disciplines have readily adopted them. DENDRAL,¹⁸ which suggests the structure of organic molecules, and R1,¹⁹ a system created by Digital Equipment Corporation, Maynard, MA, which assists in the set up of computer systems, have enjoyed broad support. The capability of DENDRAL, R1, and limited-domain medical systems such as Pathfinder demonstrates that decision support systems are feasible for routine usage in limited domains. Thus, the question remains: what must be done in order to achieve success in broader problem areas? An excellent introduction to the matter is provided by Russell and Norvig,²⁰ who point out that the field of medicine, unlike organic chemistry, lacks a general theoretical model. In particular, medical diagnosis is fraught with uncertainty. Luger and Stubblefield²¹ take the analysis further and identify five “deficiencies” of expert systems technology in general that pose particular problems in judgment-related fields such as medicine. They are summarized below:

1. Lack of “deep” (causal) knowledge of the domain (i.e., systems do not understand physiology);
2. Lack of robustness and flexibility. Systems, when faced with a problem not contained in their knowledge bases, cannot (1) solve the problem, (2) recognize their inability to solve the problem, nor (3) develop a strategy for doing so;
3. Inability to provide deep explanations;
4. Difficulties in verification;
5. Inability of systems to learn from experience.

The inability to reason with specialized data types (e.g., temporal, spatial), is another obvious shortcoming of many CDSS. The design issues mentioned thus far, that need to be addressed in order for CDSS to become more widely used, may be divided into two broad categories: (1) technical design issues, especially knowledge representation, reasoning, and knowledge acquisition and (2) human-computer interaction.

Technical Design Issues

Adding Structure to Medical Knowledge

In order to perform their desired tasks, CDSS require access to knowledge about their domains. Facts, unadorned, relate little information about the world. Meaning requires an understanding of relationships. Seeing the

number 17,000 as an isolated value carries no message. However, once it is related to a modifier, “white blood cell count,” it has clinical meaning. The goal of knowledge representation is to provide intelligent systems with information about a specific domain in a form that can be processed efficiently. The representational scheme, along with domain facts, together, constitute a knowledge base. Over the last 20 years, researchers have created a number of representational schemes ranging from simple collections of logic predicates to elaborate network structures. The expressive power of the representational scheme chosen for an intelligent system has direct bearing on the types of problems the system may be expected to solve, as well as how it goes about solving them. For example, a system for detecting and warning about potential drug interactions needs a way of representing drug classes, alternate names for medications, and the difference between drugs that are topical and those that are introduced into the body. If we were to decide later that predicting the ultimate effect of a drug on a clinical state is the desired output, temporal and physiologic information must be somehow represented in our knowledge base. Logic-based reasoning would work well for the first system; the second would require a causal mechanism.

Knowledge Representation Formats

Most knowledge representation schemes fall into one of four categories: logic, procedural, graph/network, or structured. Although not considered a classic architecture for knowledge bases, database management systems will undoubtedly play a significant role in this arena as more clinical information systems use this format for data storage. The following discussion reviews some of the concepts discussed in Chapter 2 and illustrates additional knowledge representation schemes.

Logic-Based Knowledge Representation

Propositional logic was the first representational format widely used for artificial intelligence research. As previously discussed in Chapter 2, propositions are statements about the world that are either true or false. These statements may be connected together to form sentences. Each statement may then be represented by a letter such as “P.” To illustrate, consider the two propositions “the mean corpuscular volume (MCV) is decreased in iron-deficiency anemia” and “the MCV is increased in pernicious anemia.” The first statement is represented as “P” and the second as “Q.” Propositional logic provides rules for manipulating statements. For instance, “P and Q,” “P or Q,” “P and not (Q),” are legal sentences.

The statements which we have asserted concerning the relationship between the MCV and anemia are useful; however, they must be used as whole statements, i.e., we cannot take pernicious anemia from Q and use it

to form new assertions. First-order logic (first-order predicate calculus) does offer this option. Predicate calculus provides a means of representing logic statements in a way that permits components of the assertion to be used as variables. We are no longer stuck with just “P and Q.” Using predicate calculus, the anemia propositions may be rewritten as:

MCV (increased, pernicious anemia)
 MCV (decreased, iron deficiency).

Now MCV appears as a “predicate” which provides information concerning the relationship of the “objects” it acts on (increased and pernicious anemia, as in the first example). In this form, questions can be asked of the type of MCV (x , iron deficiency), which may be read as “what is the value of the MCV in iron deficiency anemia?” This new flexibility, the ability to add predicates to a knowledge base and then to use those predicates to answer questions, provided a significant boost to the use of logic as a basis for expert system design. The programming language PROLOG (PROgramming in LOGic), which has been used to create a number of expert systems, was designed specifically to allow researchers to experiment with issues in the use of first-order predicate calculus as a knowledge representation format.

Procedural Knowledge Representations

Logic-based representations are declarative in nature in that they consist of true or false statements, and all questions are resolved through standard logic-inferencing mechanisms. In a logic-based system, the diagnosis of anemia associated with “increased” MCV would be made by looking through all the “MCV” logic predicates and finding those that have “increased” as an object. All matching predicates would then be returned (in this case, there is only one such predicate, pernicious anemia). Procedural formats, on the other hand, provide more explicit information about how the knowledge base is to be used to answer a question, it is not simply a “look up” of known facts. A procedural recasting of the anemia facts would yield:

IF MCV is increased
 THEN conclude pernicious anemia
 IF MCV is decreased
 THEN conclude iron deficiency anemia.

Notice that procedural systems offer a “process” of sorts to aid in making the diagnosis (i.e., they tell how to use the facts to draw a conclusion). These process statements are provided in the form of rules. Rule-based systems are prototypical procedural representations and have been the dominant format for medical expert systems since the days of MYCIN.¹

Networks

Networks are specialized structures consisting of nodes (representing facts, events, objects, processes, etc.) and arcs which link the nodes. As described in Chapter 2, Bayesian belief networks have proven to be very capable representation schemes for probabilistic reasoning systems, overcoming earlier objections to simple Bayesian expert systems. The flexibility of the network paradigm has greatly increased its popularity over the past 15 years. For instance, nodes in a network might consist of frames as advocated by Minsky,²² or other structures. Even more significant is the capacity of networks to capture causal, temporal, and other hard-to-model knowledge quite readily.

Decision trees²³ and artificial neural networks²⁴ are other types of network representation schemes which have recently come into favor with CDSS designers. They are discussed in Chapter 3.

Data Representation

Structural representations emphasize the “packaging” of knowledge into well defined pieces with higher levels of organization. The first widely adopted structural format was the “frame” metaphor created by Minsky.²² Frames are complex data structures which contain information about the concept being described along with procedural information detailing how the frame may change over time. For example, the concept “grocery shopping” may be represented as:

Concept: Grocery shopping
 Location: Supermarket
 Actions: Item selection (procedure)
 Paying (procedure).

Database management systems (DBMS) offer another structured format for knowledge representation. There are two types of databases which are frequently found in clinical settings—relational and object-oriented. Relational databases are based on a record structure in which each record has a number of fields. A primary field is designated, and all remaining fields in the record are related directly to this primary field. A disease record might have the following fields:

Disease Name, Organ System, Diagnostic Test, Gender Affected.

Records are then collected together into tables. Each row in the table represents a unique record and each column a feature of the record as illustrated below in Table 4.1. Additional columns could be used to improve the richness of the disease description. Each column in a relational record holds a specific type of data (e.g., number, text, etc.). However, a column cannot hold more complex data structures, for example, another record, or a list of

TABLE 4.1. Sample Disease Record.

Disease Name	Organ System	Diagnostic Tests	Gender Affected
Pneumonia	Respiratory	Sputum culture	Both
Peptic ulcer	Gastrointestinal	Endoscopy	Both

numbers. Object-oriented database management systems (OODBMS) permit greater expressiveness by permitting the storage of data types that cannot be handled by relational, table-based systems.²⁵ An anemia object might be defined as follows:

System: hematological
 Anemia Type: microcytic, hypochromic
 Disease: iron deficiency anemia
 Tests: list (serum iron, TIBC, ferritin)
 Rx: ferrous sulfate, ferrous gluconate
 Picture: (binary) peripheral smear

This anemia object contains a “list” (a collection of facts or objects) and a picture as fields in a record. More importantly, objects can “inherit” traits, thereby permitting new objects to be defined in terms of those which currently exist. This allows for the creation of new data types, a feature that is not found in purely relational systems. OODBMS have already begun to be used by researchers designing decision support systems, and they hold great promise for use in clinical information systems.²⁶

Structured query language (SQL) may be used to “ask” questions of a database, however, SQL does not support the creation of inferences, i.e., the ability to draw inferences from the data. A major drawback to using a database as a knowledge base is the lack of a specific knowledge processing mechanism for these systems. However, the ability to use higher level computer languages with database files lessens the significance of this deficiency. Still, adding inferencing capabilities is not a trivial task.

Special Data Types

Providing support for medical decisions presents unique problems to system designers because of the size of the problem domain. Adding to the situation is the need to provide knowledge about dynamic states. This requires not only facts about the objects themselves (diseases, tests, drugs, and so on), but also information concerning how these things might change over time. Predicting metastasis requires anatomical knowledge about circulation patterns and “next to” and “behind” facts. Understanding the possible effect of a medication requires knowledge of physiology (e.g., elimination times, routes, distribution). The need for causal, temporal, and spatial knowledge is a major challenge for system designers. There remains

no widely accepted format for representing the passage of time, three-dimensional anatomical relationships, nor physiological information. Programs such as ABEL, CASNET, and CHF advisor^{2,27,28} have made some progress with causal knowledge representation, however, no “portable,” generalized representation is extant.

The effective handling of temporal knowledge has been an important artificial intelligence (AI) research area from the beginning. Allen²⁹ was the first to offer a formalism for handling temporal data. He presented a format based on time points and intervals. However, the method proposed by Allen is felt to be computationally intractable when used to explain all possible relations between a set of actions and processes.³⁰ Appropriate handling of temporal information requires not only a means of representing instants and intervals, but also a formal means of representing the temporal concepts commonly used by humans. The passage of time, of which humans have an innate understanding, is not so easily represented in digital format. Basic temporal concepts such as distinguishing between future and past events, time dependency (i.e., did event X occur three minutes, three days or three years before event Y), and concurrency (while X is occurring, Y usually happens) are essential if CDSS are to reason about prognosis, outcomes, toxicities, and so on. Medical artificial intelligence researchers have created a variety of temporal representation and reasoning methods to deal with these issues. Shahar and Musen³⁰ offer a closed-interval, discrete model based upon intervals and time points. Events are represented as intervals. Intervals may have attached parameter value which can be numerical (primitive) or qualitative (abstract). There are three types of abstracted intervals: state, gradient, and rate. The RESUME system which embodies this model also includes a temporal inference mechanism and truth maintenance system, which ensures that any changes to primitive data is reflected throughout the system. TOPAZ, a system developed by Kahn et al.,³¹ which analyzes the temporal sequences of white blood cell counts and chemotherapy drug dosing, also makes use of intervals to represent temporal events in association with causal physiologic data. Kohane³² provides a third example of encoding temporal information. His experience with adding temporal information to a knowledge base points out a final issue with using temporal data. He states, “The addition of temporal information to medical knowledge bases requires significant effort. In my experience of developing modestly sized knowledge bases, . . . the task of adding temporal constraints to every event equaled that of building the rest of the knowledge base.”

All of the temporal models discussed thus far are explicit representations, whereas most diagnostic systems encode temporal data implicitly. Aliferis et al.³³ address the issue of the need for explicit temporal models and inferring mechanisms. They note that systems such as QMR, Iliad, and MYCIN manage to function quite well within their domains without having specific mechanisms for dealing with temporal data. They go on to argue

that no formal theoretical or empirical argument has been made concerning the relative value of using explicit versus implicit temporal models. It is very possible that explicit temporal modeling is more important for some types of decision-support activities (prognosis, outcomes research) than for others (diagnosis) based upon the character of the knowledge bases. Perhaps systems that rely upon frequently changing clinical data require explicit mechanisms for handling time dependent data, whereas those with more static knowledge bases and which rely on human input can perform quite well with implicit characterizations. Either way, much needs to be done in this area.

Default Knowledge

There are a number of important problems in knowledge representation and knowledge base design that are independent of format—inconsistency, degree of expressiveness, and incompleteness are ready examples. The most interesting, and without doubt, one of the most difficult to solve, is that of default or common sense knowledge. There are facts about humans which clinicians, when discussing patient problems, consider too basic to even mention—females become pregnant, men get prostate cancer. Default reasoning may be viewed as a means of dealing with incompleteness.³⁴ Consider the statement, “The patient in room 574 is 6 months pregnant.” Automatically it can be determined (by a human) that this patient: (1) needs yearly Papanicolaou smears; (2) should not receive certain medications; and (3) will have an abrupt decrease in weight in three months or so. If our knowledge base had contained the fact “males do not become pregnant” and did not have a “females become pregnant” fact, a default reasoning system might gracefully default to “the patient in room 574 is female” and proceed with its analysis. This default approach to incompleteness is not without problems. Ponder the effect of a drug dosing system that gives a medication because no statement of patient allergy is found in its knowledge base. How should default information be encoded and used? Proposed solutions to this problem will be discussed later in this chapter.

Reasoning

Due to the fact that early systems were designed by researchers interested in “artificial intelligence,” much of the work on medical expert systems was aimed at getting these systems to mimic the decision-making processes of human experts. Interestingly, programs such as MYCIN, Pathfinder, and the Leeds system, while quite capable, do not “reason” in the same manner as humans. They have no innate understanding of human anatomy or physiology, are unable to handle temporal concepts, and have no ability to learn or deduce new facts. Yet, within their narrow domain, it has been

demonstrated that they can perform comparably to human experts. However, once the domain of expected expertise is broadened, performance significantly worsens. The failure of techniques used in the design of limited domain systems to “scale up” to more general systems is a major driving force behind research in medical artificial intelligence and, by extension, CDSS. The ability to reason from “first principles” and to understand the effects of time on disease processes are considered essential to building robust systems that have more human-like capabilities. A significant amount of work has been done in the area of causal modeling (i.e., addition of anatomical and physiological data to knowledge bases).^{2,27,28} Temporal reasoning and representation have also received a good deal of attention.³⁰⁻³² Aside from adding human-like abilities, issues such as the computational burden of large numbers of calculations in networks, handling conflicting rules in knowledge bases, gracefully handling uncertainty and ignorance, and methodologies for acquiring new knowledge are sufficiently formidable so as to attract the attention of researchers. We will begin the exploration of these issues with the problem of reasoning.

Rule-Based and Early Bayesian Systems

In order to understand the research issues related to reasoning, it is necessary to trace the development of inference mechanisms in CDSS. The most basic inference mechanism utilized in medical diagnostic systems is propositional logic, which was described earlier in this chapter using the example of anemia. Another example of a knowledge base consisting of only two facts might be: “CPK-MB is increased in myocardial infarction (MI)” and “chest pain is present in MI.” All facts concerning the findings associated with myocardial infarctions would, coupled with a mechanism for testing their validity, allow one to draw a conclusion about the presence of an MI in a patient. For example, if we state as a premise that “patient X has chest pain and an increased serum CPK-MB,” it would be reasonable to conclude that the patient had an MI. This may be written in the form:

IF patient X has chest pain
AND CPK-MB is increased
THEN the problem is MI.

Notice that our small knowledge base does not contain any facts about other possible causes of chest pain; therefore, the system could not conclude that the patient has esophageal reflux. According to Russell and Norvig,²⁰ logic systems have three properties which are particularly useful. We will make use of only one of them for this discussion—“locality.” If there is a statement of the form “if a, then b” and “a” is known to be true, then we can conclude that “b” is true regardless of whatever else is known to be true. Locality is very useful in logic systems where all facts are either

completely true or completely false (for this discussion, being true is equivalent to “is only caused by” and false “is never caused by”). However, in the field of medicine, there are very few findings that can be so neatly categorized. Consider what happens when we add the fact “chest pain is present in esophageal reflux.” The presence of chest pain no longer absolutely implies MI. Locality no longer holds. Rule-based systems such as MYCIN inherit the properties of logic systems and the possibility of inconsistency in the knowledge base. The fundamental issue becomes one of handling uncertainty gracefully. Russell and Norvig offer three reasons why most systems based on propositional logic are unworkable for medical diagnosis. These reasons are related to the unavoidable presence of uncertainty and are described as laziness, theoretical ignorance, and practical ignorance. Laziness, in this instance, describes the reluctance of system designers to do the work necessary to “list a complete set of antecedents or consequents needed to ensure an exceptionless rule, and it is too hard to use the enormous rules that result (p. 463).”²⁰ Theoretical ignorance is simply an acknowledgment that there is no theory of medicine to guide modeling of the domain. Last, practical ignorance is a statement of the fact that, for any particular patient, even if we knew all the applicable rules, we would rarely have access to all the required information (tests, genetic history, etc.).

The developers of MYCIN addressed the issue of uncertainty by pioneering the use of “certainty factors”—numerical estimates of the confidence in a particular fact. The certainty factors are based upon the opinions of domain experts and are not derived from epidemiological data. Certainty factors can take on values from -1 (indicating certainty that a condition is not true) to 1 (that it is true). Zero indicates that little is known about a particular fact. This is an important feature which differentiates them from true probability estimates, which must be between 0 and 1 . For example, we could add certainty factors to our MI knowledge base:

IF chest pain is present
THEN conclude MI 0.65 (certainty factor of 0.65)

Certainty factors were an attempt to deal with uncertainty. However, as will be illustrated, their use in a rule-based, logic-derived system may lead to erroneous conclusions.

Consider the effect of adding a new rule (rule 2):

IF chest pain is present
THEN conclude esophageal reflux 0.4 (certainty factor of 0.4).

In a system where locality is expected, if rule 1 fires, then “conclude MI .65” will become the active hypothesis. Yet it is possible that rule 2 may also be valid. In order to arrive at the correct diagnosis, some mechanism must be in place to adjudicate between the two rules or the knowledge base

designers must ensure that the two rules will never conflict. In a domain such as medicine, where thousands of rules may be needed, it is easy to see how conflicts might creep into the knowledge base and undermine the accuracy of the system. An excellent discussion of the failings of rule-based systems using certainty factors may be found in Heckerman et al.¹⁴

Unlike MYCIN, the Leeds abdominal pain system was based on simple Bayesian computation. However, early Bayesian systems had their own problems. The most significant was the number of probability estimates required to make the system workable. In addition, each new piece of evidence required recalculation of all pertinent probability estimates, resulting in a burdensome number of computations. A final requirement of early Bayesian systems was “conditional independence” (an assumption that all relationships between evidence and hypothesis are independent). The inability to assure conditional independence caused Bayesian reasoning systems to lose favor with expert system developers. Thus, even though MYCIN and the Leeds system proved to be capable of performing well within their problem domains, their reasoning mechanisms were considered to be inadequate for larger problems.

Causal Reasoning

Causal reasoning, simply defined, is the use of deep domain knowledge (i.e., pathophysiology, anatomy) to assist in the decision-making process. The fact that clinicians, when faced with a difficult problem, also resort to this form of reasoning served to enhance its attractiveness as a model for inferencing in CDSS. Patil³⁵ argues very cogently on behalf of causal reasoning as a guiding principle in CDSS. He offers a few of the potential benefits—describing the evolution of diseases over time, reasoning about interactions among diseases, and the ability to understand specific mechanisms.

CASNET² was the first medical expert system based upon causal precepts. Designed to assist in the diagnosis of glaucoma, CASNET’s knowledge is represented in the system as a network of pathophysiologic states. A particularly interesting feature of CASNET is the hierarchical organization of its knowledge base. At the lowest level are patient signs, symptoms, and tests. The middle layer consists of pathophysiologic states such as corneal edema and elevated intraocular pressure. The highest knowledge level is composed of disease categories—open angle glaucoma, secondary glaucoma, and so on. Connections between the layers represent direct causal relationships, allowing diseases at the highest level to be viewed as aggregations of patient findings and pathophysiologic states. Reasoning is carried out by navigating a path from findings to disease, testing pathway nodes by calculating a likelihood value for each, then following the highest likelihood pathway.

The CHF Advisor²⁷ and ABEL²⁸ represent alternate approaches to causal reasoning. The CHF Advisor, which assists with the diagnosis and

management of heart failure, is based upon a qualitative physiologic model of the cardiovascular system. A truth-maintenance system (TMS) enforces relationships between parameters. The TMS also allows the program to test the effects of changes in a particular variable on the entire model. This permits one to experiment with the effects of altering the value of various parameters.

ABEL has acid-base and electrolyte disorders as its domain. ABEL's knowledge base, like CASNET, models its domain at three levels of detail. Its highest level represents clinical states (i.e., hypokalemia, acidosis) while the lowest level is a physiologic representation of electrolyte stores and movement between various fluid compartments. As noted by Patil,³⁵ "The critical feature of ABEL is its ability to determine and represent situations where a hypothesis is capable of explaining part but not all of an observed finding." Limited causal modeling has been used in systems such as Caduceus.³⁶ Causal links are implemented in the knowledge base in the form of "may be caused by" relationships, serving to constrain the number of nodes evaluated during the diagnostic process. The causal links in Caduceus are much more primitive than those in ABEL, CASNET, and CHF Advisor in that they do not represent deep knowledge of the domain. The use of a more superficial form of causal links has been exploited in a newer type of diagnostic system, belief networks, which will be discussed in a later section.

Causal reasoning, while effective, does have significant limitations as an inference mechanism. The lack of knowledge concerning the actual mechanism for a number of diseases remains a major impediment to the creation of causal systems—i.e., the pathophysiology of rheumatologic disorders is much less well defined than those in cardiology. Thus, general domain systems such as QMR cannot be completely built using this reasoning model. However, this does not preclude the inclusion of causal knowledge in these systems. In fact, a good deal of causal knowledge is encoded implicitly in QMR's knowledge base.¹¹

Another design issue for causal systems is level of detail. ABEL has three levels of detail represented in its knowledge base. How many should be included to be considered complete? Is a complete representation possible or even desirable? Perhaps the ultimate design issue is that of "understanding." CASNET and ABEL are designed as networks of causally linked nodes. And although they can use deep knowledge of their domains, they do not understand what they are manipulating (the "holy grail" of AI research from the beginning).

A final matter is that of temporal representation in causal networks. One of the most basic aspects of any disease is the temporal relationship among findings. If one knew that a mass has been present on a chest radiograph for 15 years, it would automatically be considered benign. What is the best way to represent the implausibility of this being malignant in a knowledge base? All of these questions add complexity to the design process.

Probabilistic Reasoning

Bayesian reasoning systems fell out of favor in the mid 1970s due to the need to develop and maintain huge probability tables (joint probability distributions) in order to perform the required calculations. Aside from the need to maintain probability distribution data, a separate and equally daunting problem was that of assuring conditional independence of findings. In many cases, especially for any sufficiently large domain, this was difficult to achieve. A solution to both problems was advanced by the findings of a number of researchers in the form of belief networks.^{14,37,38}

A belief network is a directed acyclic graph (the arrows point in one direction and there are no circular paths) consisting of nodes that contain conditional probability data. Nodes may be thought of as “parent” and “child,” with parent nodes connected to child nodes by one-way arrows. Conditional probability tables at each node reflect the effect of all the parent nodes on the child node. As might be expected, even this format is computationally intensive for all but small networks. If you will recall, one of the problems of early Bayesian systems was conditional independence. This criticism is addressed by network designers via the use of causal relationships when creating networks and by use of a catch-all probability estimate in the form of a “noise parameter.” If we say the probability of MI is 0.7 given finding X, and 0.2 given finding Y, and if 1.0 represents certainty, then 0.3 represents the noise for X and 0.8 for Y. Conditional independence values are not exact, but the use of noisiness permits usable systems to be built.³⁹ Heckerman et al. achieved an acceptable solution by building a limited number of conditional dependencies into their Pathfinder Network.¹⁴ Judea Pearl has produced an excellent text on this subject for those who wish to develop a fuller understanding of this area.⁴⁰

Decision-Theoretic Reasoning

A somewhat newer approach to medical expert systems design is the use of decision theory in the reasoning process.^{41,42} Decision theory is based upon the concept of utility—the value to the decision maker of a particular outcome. In the case of a patient with chest pain where either esophageal reflux or MI might be the cause, a pure probabilistic system would offer as its conclusion the diagnosis with the highest probability (for the sake of argument assume that this is reflux). In a decision-theoretic system, the cost to the patient of suggesting reflux when the correct diagnosis is MI would be calculated before offering a final conclusion. Thus utility serves to “remind” the system of the “cost” of an incorrect diagnosis or suggested action. A significant problem with decision-theoretic systems is that of determining how the utilities included in a system will be determined—

never a simple undertaking. Nevertheless, this is a promising development in the design of decision support systems.

Possibilistic Reasoning

The discussion of reasoning thus far has focused on the handling of uncertainty. Uncertainty is an expression of the inability to know all the factors involved in a particular decision and their ultimate effect upon the outcome. Lofti Zadeh pointed out another decision-making dilemma—imprecision in the expression of a finding or factor.⁴³ The statement, “cervical cancer is a disease of younger women” is an example of fuzziness. At what age does a woman stop being young? Zadeh proposed that unlike traditional set theory in which an element was in only one set, membership may be possible to some extent in a number of sets. Thus a 35-year-old woman would have partial membership in the old set (say 0.3) as well as in the young set (0.7). Fuzzy logic provides a formalism for computing the truthfulness of fuzzy propositions.⁴⁴ Maiers has written a good review of fuzzy logic in medical expert systems.⁴⁵

Accounting for Ignorance

Shafer⁴⁶ describes the Dempster-Shafer theory of evidence as a means of dealing with ignorance, as opposed to uncertainty. This theory arose out of the difficulty of assigning prior probability values. In most situations, these values are estimates and therefore subject to error. The Dempster-Shafer theory proposes that probability estimates be qualified by using a “belief function” that computes one’s belief in a particular proposition. Belief functions add to the computational complexity of a system and, due to their weaker theoretical grounding (as compared to Bayesian and fuzzy systems), have received less support among CDSS designers.

Common Sense Reasoning

Common sense reasoning, at its most basic level, is about making assumptions. This is an indispensable capability that we use constantly. Common sense (default) reasoning allows objects to be grouped into recognizable classes that can be mentally manipulated based on common traits. For example, “birds fly, cars use gasoline, and planes land only at airports” are statements about classes of objects that are well defined. If one is then told that a car would not run, an automatic question would be whether it is out of gas. Now consider what happens when decisions have been made using the facts mentioned previously and then it is discovered that electric cars and emus exist. What should be done? Should all prior decisions be revised?

How should these new facts be added to the knowledge base? New facts concerning objects currently represented in the knowledge base must be reconciled with those already present. How should precedence be determined? Also, how should conclusions added to the knowledge base under the influence of old rules be updated or retracted? Default logic as suggested by Reiter³⁴ and the nonmonotonic logic of McDermott and Doyle⁴⁷ are offered as reasoning mechanisms to deal with these problems. However, no system is available which adequately addresses all issues. In addition, no working CDSS have been designed using default reasoning mechanisms.

Case-Based Reasoning

Case-based systems offer an approach to learning and reasoning that is very different from those discussed previously. A case, as defined by Kolodner and Leake,⁴⁸ is “a contextualized piece of knowledge representing an experience that teaches a lesson fundamental to achieving the goals of the reasoner.” Case-based knowledge bases have two distinct parts: the case itself and an index that aids efficient context-based retrieval. Case-based systems acquire knowledge by solving problems. Cases are stored knowledge that reflect past experience in solving problems. Each case has three components—problem/situation description, solution, and outcome. The problem/situation-description describes the past situation or problem that was solved. It includes the goals of the reasoner as the problem was being solved, as well as information about the problem environment. The solution component contains information regarding how the problem was solved. The result of applying the solution, whether the attempt succeeded or failed, and why, are stored in the outcome component. Access to cases is controlled by an index. The key to solving problems in case-based systems is matching the current problem to past experience. Compared to more traditional approaches, advocates of case-based systems believe that these systems have the following advantages: (1) they are better at solving problems with open-ended, poorly defined concepts; (2) they arrive at solutions faster; (3) they are better at solving problems where no good algorithm is available; and (4) cases may serve as explanations.

Case-based reasoning is not without problems, however. In a large knowledge base, retrieval efficiency is an important determinant of performance. Therefore, indexing is a key research area. Issues such as whether to use high-level or low-level features when building indexes, how to design a general framework for index content, and the design of case retrieval algorithms remain a source of vexation for those designing case-based systems. Despite its problems, case-based reasoning has been used successfully in CDSS^{49,50} and offers an interesting metaphor for building flexible knowledge-based systems.

Knowledge Acquisition

Knowledge Engineering

Knowledge engineering is the process of building a knowledge base. A knowledge engineer is a professional with an understanding of issues in knowledge representation, tool selection, artificial intelligence languages, and software design. A knowledge engineer works with a “domain expert” to obtain the necessary data to build a knowledge base (knowledge acquisition). This has been the traditional model for building expert systems. Knowledge engineering can be a very tedious process. The time required to build a knowledge base for any decent-sized domain is often considerable and greatly inhibits the production and deployment of knowledge-based systems. Much of the difficulty in building expert systems in any domain is due to the lack of a well defined process for the activity. Even with the availability of specialized shells, languages, and other tools, the knowledge extraction process is often still haphazard. Domain experts can be very poor at describing what they do or how they approach a problem. The knowledge engineer often has to learn the domain in order to identify major unifying concepts. Next, the actual problem to be solved must be agreed upon, and, finally, knowledge representation formats and a reasoning mechanism must be chosen. Errors made at any step can result in significant delays and frustrations. Once completed, maintenance becomes a serious problem which can worsen with turnover of the development team. The “knowledge acquisition bottleneck” has no real solution using traditional methods. Also, standard knowledge engineering practices do not take advantage of the large amount of data stored in information systems currently in use.

Managing Knowledge Using Ontologies

In most instances, knowledge bases are collections of facts about the real world, encoded in a manner that allows them to be used computationally. However, more sophisticated decision support systems will require access to not only facts, but also key concepts that underlie the domain. For example, accounting for the passage of time or the flow of blood through specific arterial pathways requires deeper knowledge of the world than simple assertions that a disease is chronic or that a coronary artery is obstructed. The field dedicated to creating these higher-level conceptual maps is ontological engineering, and the resulting knowledge construct is an ontology. Gruber,⁵¹ has defined an ontology as “. . . a formal, explicit specification of a shared conceptualization.” Thus, ontologies are viewed as resources that may be used by people or computers for three basic purposes: communication, computational inference and reuse, and knowledge management^{52,53}. Communication is enhanced by the requirement that all terms and concepts be standardized and clearly defined, providing for a

common reference point for all ontology users and builders. As concepts and terms are defined, all relationships that they may participate in are likewise defined along with the appropriate access algorithms and rules for the proper use of all terms, concepts, and relationships. Computational use of the knowledge contained in the ontology is thereby codified, making it easier for software designers to understand how to interact with the knowledge base to build intelligent systems. Finally, the framework created in building an ontology provides for a high-level mechanism and viewpoint for extending and maintaining the knowledge contained therein, which greatly simplifies key knowledge management tasks.

In order to function as a viable resource for an organization or any group of users, “ontological commitment” (all parties who use the ontology agree to abide by its concepts, terms, and relationships) is a must.

Ontology Organization

Ontologies seek to represent the world (or at least a specific domain) in an organized manner. Objects, events, and processes that occur in the real world are captured in a standardized manner based upon fundamental concepts taken from the domain. The highest level (most abstract) of concept organization is referred to as an “upper ontology.”²⁰ The upper ontology provides the basis for all concepts that will eventually make-up the final operational ontology. Figure 4.1 below represents a very simple upper ontology of living things.

Concepts are rendered in computable form as categories (templates/classes) that have specific properties. Upper ontology categories are abstract entities; as such, they do not refer to a specific person or plant. Rather, the most fundamental properties of each are defined at this level (with special attention given to those properties that separate one category from another). A specific person or plant can then be represented as an object that inherits (“is-a” relationship) properties from all the categories from which it was derived. Inheritance that occurs via hierarchies may be

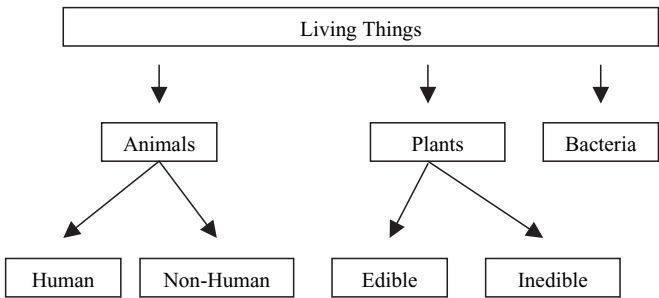


FIGURE 4.1. An upper ontology of living things.

single (derived from a single upper ontology concept) or multiple (derived from more than one high-level concept). Management of the complexities of multiple inheritance has tended to favor the creation of mainly single inheritance ontologies. Therefore, specification of a single person, “John Doe,” might be accomplished by inheriting key properties along the “Living Things: Animals: Human” hierarchy.

A second mechanism for enriching the representational capability of ontologies is that of associations. Associations permit links to be formed between classes and define allowable interactions between the two. For example, a “uses for food” association might be defined between the “Humans and Plants: Edible Classes.”

Events (birth) and processes (movement) can be represented using this conceptual organization as well. The key is that all those who use the ontology must agree to the exact meanings and proper usage of all concepts, terms, and objects.

Examples of Ontology-Based Systems

Though not yet common as knowledge resources for clinical decision support systems, excellent examples of systems that use ontological methods for representation of biomedical concepts do exist. Perhaps the most frequently used is found within the Unified Medical Language System (UMLS).^{54,55}

UMLS Semantic Network

The UMLS is an ongoing project of the National Library of Medicine as part of its efforts to improve access to reference resources. The UMLS acts as a link between a number of disparate vocabularies and coding systems (e.g., ICD-9, CPT, SNOMED) and the Medical Subject Headings (MeSH) coding system. Its major components are the Metathesaurus, which integrates terms and concepts from over 60 code/vocabulary systems, and the Semantic Network, which provides a high-level conceptual framework for categorizing terms found in the Metathesaurus. The upper ontology of the UMLS semantic network has two basic concepts: Entities and Events. The next level below describes Physical Objects, Conceptual Entities, Activities, Phenomena and Processes. Multiple semantic types are defined along with types of associations: physical, spatial, temporal, functional, and conceptual.^{56,57}

GALEN

GALEN (Generalized Architecture for Languages, Encyclopedias, and Nomenclatures) is designed to act as a terminology for use in clinical systems⁵⁸ GALEN is managed by OpenGALEN, a Dutch-based nonprofit foundation, which provides access to Galen technology, and has been used

in Europe to create coding systems for surgical procedures. GALEN is an ambitious project in that it intends to encompass all of medicine and has as its major goal the creation of a terminology that is both computable and usable by humans for direct data entry.

The upper ontology of GALEN has two categories: Phenomenon and ModifierConcept.⁵⁹ Phenomenon subsumes Structures, Substances, and Processes, providing classes for representing entities that exist as objects in the real world (e.g., organs, people, drugs). ModifierConcepts provide classes for representing features of real world objects (e.g., units of measure, degrees of severity, etc.). Individual terms in GALEN are derived via use of “part-of” associations (e.g., component-of, stuff-of, portion-of, member-of) links that can be inherited.

All GALEN concepts are modeled in GALEN Representation and Integration Language (GRAIL), a formal language system developed for the GALEN Project.⁶⁰ GRAIL provides compositional capabilities to GALEN (i.e., ability to form complex concepts using simple primitives) and supports generation of semantic networks. GALEN concepts are embodied in the GALEN Common Reference Model and access to concepts is managed through the GALEN Terminology Server. It is the terminology server that application developers interact with in order to access the vocabulary.

SNOMED CT

SNOMED CT⁶¹ is a terminology based on SNOMED RT⁶² and Read Codes Version 3.⁶³ SNOMED is a comprehensive clinical terminology encompassing 344,000 concepts and close to 1.3 million semantic relationships. The upper level ontology of SNOMED CT has as its top node Root-Concept.⁶⁴ Below this node are TopLevelConcept Nodes: 13 that define clinical concepts and three that provide structural concepts for constructing relationships, assigning values, or other internal functions such as concept navigation. The top-level clinical concept nodes define concepts for: Findings, Disease, Procedure, Observable Entity, Body Structure, Organism, Substance, Pharmaceutical/Biological Product, Specimen, Physical Object, Physical Force, Events, Environments/Geographical Locations, Social Context, Staging, and Scales. Top-level structural nodes define: Attribute, Qualifier Value, and Special Concept entities. SNOMED provides for inheritance (is-a) from multiple concepts, and associations between concepts. Complex concepts may be constructed using simple primitives.

There are efforts underway to key nursing, drug, and administrative terminologies to SNOMED CT, making for a robust clinical terminology.^{65,66}

Ontology Issues in Decision Support

The ready availability of high quality tools, such as Protégé,^{67,68} for creating ontologies, bodes well for those creating decision support systems. Share-

able knowledge bases may be created, that could be used in any number of systems. Certainly within large integrated delivery systems, software developers could produce knowledge bases that permit terms and concepts to be shared by key technologies such as electronic health records or research database systems. Of course, the more complex the decision to be made, the more difficult it becomes to build an appropriate computer model of the domain to support it. Systems that provide straightforward alerts and reminders are easier to design and build than those which support more complex decision tasks such as automation of complex practice guidelines.

The Frame Problem

Complex decision support activities, such as multistep guidelines with steps that occur over time, have to contend with the problem of changing data. The “frame of reference” problem,²⁰ keeping track of what should and should not change after some key event, is a major challenge. An example of this would be assuming a decision is to be made, that requires checking a lab value and then, based upon that value, some series of interventions is suggested. Each suggested action may change the underlying system (i.e., patient) in ways that make it difficult to predict what the ultimate effects will be. There are two problems: keeping track of how the patient has changed and what has remained the same. Humans quite naturally reason out these “what-if” scenarios; computers require large knowledge bases with complex rule sets to mimic this type of reasoning.

Lack of Standards

Ontology tools have evolved largely as experimental technologies over the last 10–15 years and, as a result, they do not conform to a particular standard in terms of internal representation format, interface, or language.^{53,69,70} A key issue in building any ontology is term selection. Naming things is very important, and sharing an ontology within a domain requires that all users share a common terminology. SNOMED CT appears to be the most promising candidate for health care. However, it is far from being widely used for key terminology needs within the United States. Also, one cannot mistake agreement on terms for agreement on definitions. For example, two hospitals may agree to use common terms to pool data. They settle on a SNOMED CT term, *Congestive Heart Failure*, to represent a key concept for their disease management program. Hospital A permits the use of this term for any patient with symptoms of orthopnea and pedal edema, whereas Hospital B requires a left ventricular ejection fraction of less than 40% to use the term. Obviously, even though they have agreed on a specific term, the definitions are not the same. Hospital B has a much better defined pool of patients and, if one were analyzing outcomes of CHF treatment, the data from Hospital B is likely to be of higher scientific value.

Ontologies promise to be the next major advance in knowledge representation and management. However, they are by no means easy to design or build, and their ultimate value will be determined by the willingness of healthcare organizations to commit to (1) very specific ways of accessing and using the knowledge and (2) putting forth the effort to clearly define key concepts, actions, and terms—something for which there is little precedent.

Capturing Clinical Detail—Codes, Classifications, Nomenclatures, Vocabularies

Using computers to manage data requires some means of capturing information in a way that permits all potential consumers of medical data to use it effectively. The Institute of Medicine in its 1992 report on the computer-based patient record⁷¹ detailed the problems of the paper record, in terms of content and ability to aid in decision making. The issues of content and utility that plague paper records must be addressed to realize the decision support benefits in electronic systems. The capture of detailed, analyzable clinical information in electronic form is a well recognized problem in the field of medical informatics.^{72,73,74} Though a number of terminology sets exist, none has been found that satisfies the wide range of data requirements in the broad domain of health care. Adding to the problem is the lack of a standard definition for the various terminology sets.^{75,76,77,78} Is it reasonable for anyone new to the field to be completely baffled in trying to determine the difference between a code, classification, nomenclature, and controlled vocabulary?

Van Bommel and Musen⁷⁹ offer working definitions that may help to guide this discussion. They define classifications as “. . . an ordered system of concepts within a domain, with implicit or explicit ordering principles (p82).” The most common classification in use in the United States is the International Classification of Diseases (ICD). It began in 1893 as the Bertillon Classification or International List of Causes of Death, and was used as an epidemiological tool.⁸⁰ In ICD version 9, the most used version containing nearly 7,000 entries, the terms may code for diagnoses (e.g., myocardial infarction) or symptoms (chest pain). The ICD is arranged along a variable axis and continues to order codes based on a structure suggested by William Farr⁸¹ using groupings of epidemic diseases, constitutional or general diseases, local diseases arranged by site, developmental diseases, and injuries. The entire classification is divided into 21 chapters (e.g., Chapter II: Neoplasms, Chapter III: Hematological diseases) with a strict hierarchy followed within each chapter. Each term is given a three-digit code with a possible two-digit extension (e.g., diabetes mellitus—250.00). The most recent version, ICD-10, has been updated to include more than 13,000 terms and has been extended to cover additional health-related problems.

An obvious problem with the ICD system is the lack of clinical detail. Systems designed for monitoring epidemiologic trends lack the granularity needed for clinical care.^{82,83,84} A decision support system that relied on ICD codes for suggesting interventions would be reduced to making only the most general advice concerning possible actions due to the paucity of specific clinical information about the patient.

Codes are means to reduce the amount of information required to communicate a concept. Thus, the ICD codes represent its classifications. There are no rules regarding how coding schemes are to be derived; it is up to the designer to select a representation format most suited for the purpose at hand. The Current Procedural Terminology (CPT),⁸⁵ published by the American Medical Association since 1966, was designed to help with reimbursement for medical services. It consists of approximately 8,000 codes that are five digits in length, with intervention name and brief description. Codes may be retired or replaced with newer ones. Like the ICD, CPT codes do not capture any level of clinical detail aside from the name of the intervention performed. The CPT is not arranged in a hierarchy, and all interventions that it encodes are grouped by type (microbiology, chemistry, radiology, etc.). The CPT and ICD systems are the most commonly used code sets in the United States and are required for many administrative tasks.

A recent addition to the library of available code sets is the Logical Observation Identifiers Names and Codes (LOINC) set.^{86,87,88} LOINC provides a formal multi-axial code set for laboratory results and clinical observations. A total of about 32,000 results and observations are included. LOINC codes are never removed from the system, greatly improving the ability to maintain accurate longitudinal data. LOINC entities may have entries along up to six axes: Component, Property, Time Aspect, System, Scale, Method. LOINC provides a significant amount of clinical detail and is much more readily suited for clinical decision support tasks than either the ICD or CPT.

A nomenclature is defined as a system of words used in a particular discipline. A key distinction between nomenclatures and codes/classifications systems is that the former permits one to form new concepts by combining terms, whereas the latter do not. SNOMED CT is the latest iteration of the SNOMED nomenclature, first published in 1975, and its recent endorsement by the National Health Service in the United Kingdom and the Department of Health and Human Services in the United States should lead to its use in newer clinical decision support systems.

“Clinical terminology” and “controlled clinical vocabulary” are somewhat more difficult to define. The terms are often used interchangeably by many, and there is some disagreement as to the content and scope of the technology that each term represents.^{75,76,77,78} Generally speaking, it would seem that a controlled vocabulary may be derived from any of the above systems. The key is that the use of all terms is closely constrained (e.g., users cannot introduce novel terms into the system). This may be done either

through the use of a restricted list (e.g., in a combo box on a form), or by the application of algorithms for term selection. Even without a widely accepted definition, most views of controlled clinical vocabularies are that they are highly granular term sets arranged in well defined hierarchies that are designed to support the capture of all aspects of clinical care. It is this view of expressiveness and utility that appears to underlie the analysis offered by Cimino in his landmark article, “Desiderata for Controlled Medical Vocabularies in the Twenty-First Century.”⁸⁹ While no explicit definition is offered, the author does provide a very practical approach to understanding and analyzing the requisite features, functions, and architecture for any terminological system intended for the capture of highly granular, longitudinal clinical data. The desiderata consist of guidelines by which controlled clinical vocabularies may be judged. Cimino’s criteria address issues of design, management, and operational use of vocabularies, providing very helpful guidelines for those whose wish to make use of these technologies. (See Table 4.2 for a summary of these criteria).

Vocabulary Utilization Issues

Using Cimino’s criteria, it is easy to see that ICD, CPT, and other code sets might be quite useful for tasks that do not require a great deal of detail, while more robust term sets are required to capture clinical detail. Of course, the availability of a controlled clinical vocabulary and integrating one into a clinical application are two very different issues.⁹⁰

Spackman and colleagues,⁶² have identified three types of clinically relevant terminologies: application, user interfaces, and reference. Reference terminologies are defined by them as “. . . a set of concepts and relationships that provides a common reference point for comparison and aggregation of data about the entire health-care process, recorded by multiple different individuals, systems, or institutions.” Interface terminologies are those that are encountered by the user during data entry. Here, the goal is to present, as quickly and clearly as possible, a collection of acceptable terms for data entry. This requires hiding a good deal of the complexity of the underlying reference terminology from the user while allowing for acceptable data entry speed and expressiveness.

Data Entry Concerns

Completing a problem list or ordering laboratory tests are simple data entry tasks, easily accomplished with pick-lists. However, entering a complete history and physical examination for a typical primary care encounter in this manner would quickly become quite tedious. Solving the problem seems to require a trade-off between efficiency and expressiveness.^{91,92} For example, templates might be used to increase data entry speed but may have the unintended effect of homogenizing clinical descriptions with an

TABLE 4.2. Summary of Cimino's Desiderata.

Content	Sufficient terms should be present that permit the entire spectrum of clinical concepts to be clearly represented.
Concept-based	All terms must have at least one meaning and only one meaning, and any meaning is assigned to only one term.
Concept permanence	Once created, concepts are never purged from the system, permitting linking of newer concepts to their progenitors.
Polyhierarchy	Concepts should be bound to terms following a specific hierarchy and, when necessary, may belong to multiple hierarchies. Pneumococcus may be found by following the "Organism" tree or the "Infectious Diseases" tree.
Nonsemantic identifiers	The unique identifiers that are assigned to concepts should not follow any particular hierarchy or organization in which the codes themselves have meaning. Example: all cardiovascular concepts should not have identifiers that code heart failure as CVS1.0, right heart failure CVS1.1, high output failure CVS1.01, and so on.
Formal definitions	All concepts and terms should be clearly defined and, ideally, definitions are stored in a computable form.
Reject "not elsewhere classified"	Concepts that depend, for their meaning, on the exclusion of all other terms create semantic problems in historical data and should not be allowed.
Multiple granularities	Different users of a vocabulary may wish to capture varying levels of detail (epidemiologist versus an internist). The vocabulary should not dictate the level of detail used to record a datum.
Multiple consistent views	Just as different users may desire varying levels of granularity, so might they also prefer to display, or provide access to, terms and concepts at different levels of detail.
Context-sensitive representation	Chest pain may be a symptom or diagnosis. Application designers should provide the context for term interpretation, not the vocabulary.
Evolve	A well designed vocabulary should be able to manage the addition of new terms and concepts to match changes in medical knowledge and practice without disrupting current users or altering the meaning of historical data.
Recognize redundancy	Synonyms for concepts are acceptable (if they all map to the same root concept). However, there should not exist multiple primitive representations of the same concept. Example: "myocardial infarction" and "heart attack" should exist as a concept and linked synonym, not as two separate concepts.

attendant loss of detail. Another approach might be the development of more intelligent interfaces that can anticipate term requirements and offer intelligent guesses during data entry. However, this could add greatly to the complexity of the programming logic of the underlying application. Validation of user input cannot be ignored.

The ability to compose complex concepts from the juxtaposition of two or more atomic concepts, referred to as a “compositionality,” greatly increases the expressive power of any vocabulary but brings with it the problem of how to prevent the creation of nonsense concepts (juxtaposition of terms that make no clinical sense).^{60,93,94} In practice, it would require that either the applications have extensive knowledge of medical concepts and language or that additional technologies must be used to provide the required functionality. The creators of GALEN address this problem by provision of the Galen Terminology Server as a basic GALEN technology. Terminology servers^{95,96} aid in the deployment of vocabularies by providing services to application developers such as lexical matching, word completion, terms composition, and other key functions. It seems likely that complex software such as electronic health records would have extensive terminology requirements and would require some level of local control of term management (e.g., only for problem lists, all data entry, specific specialties, only in research settings). Even with access to professional terminology management tools and or terminology servers, organizations could find themselves with unexpected costs associated with vocabulary management. Obviously, providing users with responsive, helpful vocabulary services is not as simple as licensing a clinical terminology.

An additional interface issue is that of local terminologies. In health care there are often many ways to say the same thing. Permitting users the freedom to use familiar terms and phrases is a reasonable system design goal. However, providing this flexibility in the presence of a controlled terminology can be tricky. Providing access for local terms may make data entry friendlier; however, it makes application design more difficult because a mapping of local terms to the reference terminology is required. If there are but a few local terms, then the problem is very manageable. However, with a sufficiently large local term set, mapping and maintaining an interface terminology and reference terminology could easily become quite burdensome. There is also the possibility that, by creating local term sets, some level of “semantic drift” will occur between sites using the same reference terminology, defeating much of the value of having a standard reference terminology. How will local term sets be managed? Getting different clinical specialists to agree on a standard term set is not likely to be a simple process, especially if the terms become mandatory for all data entry. Having a terminology server would make the ability to map local terms to a reference terminology manageable; however, it creates two new problems: (1) how to manage the local terms and keep them properly aligned with the

reference terminology, and (2) how to prevent “semantic drift” across non-related sites.

Maintaining a local terminology for an integrated delivery network could easily become a costly endeavor in terms of the talent and time to manage the terminology, unless extremely powerful and easy to use tools are available. The mapping problem alone could require an inordinate amount of time from both technical and clinical staff. A major reason often quoted for the development of standards is to promote data sharing. Healthcare entities that create local term sets could easily map similar concepts to very different reference terminology concepts or vice-versa. In either case, the data will no longer be sharable due to semantic drift. Even worse, it may not be obvious that any disparities exist.

Storage and Retrieval of Encoded Data

Compositionality creates a second design problem for system designers at the database level. Complex concepts (e.g., myocardial infarction) that exist atomically in a terminology make it easy to determine how to record the concept within a clinical information system—one simply selects the code for the concept. This is known as “precoordination.”^{97,98} Precoordination decreases complexity at the interface and database levels because there is no question as to the proper way to encode a concept. All users, no matter where they are located, are likely to encode the same concept in the same manner. When no atomic concept is available, then complex concepts must be created using collections of atomic concepts (postcoordination).^{97,98} Postcoordination provides expressive power; however, in the process it makes it easy to encode the same complex concept using different sets of atomic concepts. Even in situations in which the same codes are used, there is no mandatory order in which they must be sequenced. In practical terms, if four codes are used to encode a concept, then there are 24 unique sequences in which they may be stored, making database retrieval quite problematic even within the same organization. Consider the damper this would place on pooling data from different entities. Unless a central authority dictates all terms for all concepts, it will be quite easy (and very likely) that different organizations will create incompatible postcoordinated terms for the same clinical concept, thereby reducing the portability and pooling of data for analysis—one of the main justifications for having a standard clinical terminology.

The availability of a standard clinical terminology is an absolute necessity for the capture of detailed clinical data. However, there is no mechanism in place that will prevent developers and healthcare entities from using them in nonstandard ways. Reasonably complete terminologies are so new that many questions regarding their use in application development, and at the user level, remain unanswered.

Human–Computer Interaction

It is now time to discuss the issues raised by Engle⁹ and others concerning the lack of widespread use of decision support technology in clinical medicine. Heathfield and Wyatt¹⁷ provide an analysis of what they consider to be the psychological and organizational barriers that explain the lack of use of these systems. Their opinion is that most systems have not been designed to address the problems that clinicians actually face. Several systems are designed to restrict the number of active diagnostic hypotheses (which doctors do quite well), while few are designed to help with differential diagnosis and treatment advice. The latter types have been much better received by physicians than the former. Identifying a well defined problem to solve should be, but oftentimes has not been, an important consideration for systems designers.

CDSS must take clinicians' work habits into account. Systems must be available at the point of care and should be easy to use if they are to be considered clinical tools. The criticism offered by Clayton and Hripcsak¹⁶ extends this analysis by noting that stand-alone systems which require significant data entry will not be used on any regular basis. Finally, the single-problem focus of many systems means that they will be needed only on rare occasions, at which time it may not be worth the trouble to locate and use them. This could even be true of general systems covering multiple domains which, as Miller and Geissbuhler note in Chapter 5, might be justified in only a small percentage of patients.

Problem Knowledge Couplers (PKCs), as advanced by Weed,⁹⁹ represent a rather unique approach to the use of diagnostic/therapeutic decision support. PKCs are intended to be used at the point of care and on a regular basis, not just for cases that are perceived to be diagnostically difficult. In fact, they are designed so that even nonmedical personnel can enter the patient's data, although the physician must still interpret the output. PKCs consist of a knowledge base of diagnoses, findings, and management options. Each individual coupler addresses a single presenting problem. The couplers permit controlled input of findings and guide the clinician in the process of diagnosis and management.

PKCs are interesting from the standpoint of user interaction because they are meant to be an integral part of each clinical encounter. However, their use has not become widespread. The use of PKCs represents a significant intrusion into the clinical practice environment, and it is not clear that they will be useful for a large number of patients. For example, consider having a coupler for headache diagnosis and management. For most patients, an experienced clinician can easily differentiate between types of headaches based on clinical presentation. Using a coupler would be expected (as with other broad-domain systems) to be helpful for only a small number of cases. Thus the return for time invested, if used for every headache patient, would be very low.

Heathfield and Wyatt¹⁷ have identified several other problems systems designers need to avoid. The first is a preoccupation with computer artifacts. In other words, when beginning a project there may be a tendency to focus more on what language to use, hardware configurations, and development environments, than on the problems of potential users. This preoccupation can hamper the quest for the best problem-solving process and techniques for solving the particular problem. Next, they argue that systems designers may fail to use appropriate models for solving problems and may fail to communicate clearly the design issues to potential users.

The final problem mentioned by Heathfield and Wyatt is that designers sometimes focus on system development and ignore organizational issues. Organizational attitudes and support play a critical role in the development and implementation of any technology. The multidisciplinary nature of CDSS development makes this process even more vulnerable to problems of changing personnel, funding, administrative buy-in, and shifting organizational goals. Successful deployment of CDSS technology requires that all these matters be addressed via specific organizational policies for the creation and utilization of knowledge-based tools.

Fortunately, many of the criticisms of CDSS are readily addressed by the growing use of electronic health record systems (EHRS) and computer-based physician order entry (CPOE) systems. EHRS solve many of the CDSS problems related to work flow, data entry, and types of decision support provided.^{100,101} They provide a standard interface for users and a data model for CDSS designers. Access to laboratory, pharmacy, and other standard data is a key feature of EHRS, and allows CDSS designers to shift the focus from data entry to data access and user interaction. Alerts and reminder systems have proven to be effective for preventive care, error reduction, and patient safety.^{102,103} However, more sophisticated decision support functions, such as automation of complex guidelines, will require advances in EHRS design. Automated guidelines that contain multiple steps and act over multiple patient encounters must address issues related to the frame problem and must have access to well structured conceptual knowledge about patients and disease states. Thus, a robust ontology and controlled vocabulary are minimum requirements. Today, no EHRS has these features, and until standards for EHRS are clearly defined, systems are unlikely to appear that support sophisticated knowledge-based functions.

CPOE systems have made a mark in hospital settings where, due to the range of legacy systems present, implementation of packaged EHRS software would be problematic. While CPOE systems do not have the level of integration at the data level that an “all-in-one” EHRS provides, they are effective within their limited area. However, like EHRS, their utility is limited by the quality and completeness of the underlying knowledge base.

Interestingly, with the advent of EHRS and CPOE systems, user interface issues do not disappear—they simply change focus. For example, an

EHRS or CPOE system that provides drug interaction checking during prescription writing offers this function as a background process—the user does not have to explicitly invoke or shut down the feature. However, if there is no ability to alter the type (warn for only severe interactions) or frequency of advice (only for new medicines, not refills) users may become reluctant to use the system. Similarly, when complex automated guidelines become available, the challenge to system designers will be not only how to provide advice seamlessly, but also how to quietly recede into the background and allow the clinician to proceed with the task at hand. In fact, we may find that the ultimate issue in human–computer interaction is not one of functionality, but sovereignty. That is, once the challenges of workflow, responsiveness, and ease of use are addressed, there will remain the problem of humans who do not wish to take advice from a machine.^{104,105,106} Perhaps there is a “threshold of intrusiveness” that, once exceeded, will result in clinicians ignoring any advice provided.

Conclusion

The history of artificial intelligence in medicine is a mixed one of impressive creativity coupled with limited successes, small gains, and, in the case of Engle, cynical resignation.⁹ However, the increasing deployment of electronic health records and computerized order entry systems provides fertile ground for offering next-generation decision-support functionality. Nevertheless, serious challenges remain. Our incomplete understanding of the clinical reasoning process, and lack of an all encompassing “theory of medicine” will continue to be both sources of consternation and wonderfully intriguing research problems. The work done on knowledge representation, reasoning mechanisms, machine learning, and knowledge acquisition has wide applicability and many potential benefits for society, even if no truly intelligent clinical decision support system is ever built. Ontological engineering, and clinical terminology design and deployment, are poised to become major areas of theoretical and applied research in medical informatics as the quest for sophisticated decision support systems such as automated, multistep guidelines moves forward.

Do doctors really want help with management of clinical care? Certainly the enthusiasm demonstrated by policymakers for technology as the key to addressing patient safety and quality improvement concerns is not matched within the practitioner community. The growing acceptance of EHRS and CPOE technology would seem to indicate that the reluctance of clinicians to use computers was as much a function of less than acceptable user interfaces, lack of consideration for workflow, and the high cost of ownership rather than innate contrariness. However, it may well be that current systems, which use simple decision support mechanisms, are not sufficiently intrusive to cause a backlash.

One thing is certain: should all the problems associated with the design and implementation of CDSS ultimately be solved, to gain wide acceptance they must provide decision support without violating two of the most fundamental social and intellectual features of the practice of medicine. First, they must not intrude on the sanctity of the patient-physician relationship. Second, they must do nothing to remove or alter the quiet satisfaction derived from knowing that one has made a difference.

References

1. Shortliffe EH. Computer-based medical consultations: MYCIN. New York: Elsevier; 1976.
2. Weiss SM, Kulikowski CA, Amarel S, Safir A. A model-based method for computer-aided medical decision making. *Artif Intell* 1978;11:145–172.
3. de Dombal FT, Leaper DJ, Staniland JR, McCann AP, Horrocks JC. Computer-aided diagnosis of acute abdominal pain. *Br Med J* 1972;2:9–13.
4. Payne TH, Hoey PJ, Nichol P, Lovis C. Preparation and use of preconstructed orders, order sets, and order menus in a computerized provider order entry system. *J Am Med Inform Assoc* 2003;10:322–329.
5. Kaushal R, Shojania KG, Bates DW. Effects of computerized physician order entry and clinical decision support systems on medication safety: a systematic review. *Arch Intern Med* 2003;163:1409–1416.
6. Foster RA, Antonelli PJ. Computerized physician-order entry: are we there yet? *Otolaryngol Clin North Am* 2002;35:1237–1243, vii.
7. Shu K, Boyle D, Spurr C, et al. Comparison of time spent writing orders on paper with computerized physician order entry. *Medinfo* 2001;10:1207–1211.
8. Ash JS, Gorman PN, Seshadri V, Hersh WR. Computerized physician order entry in U.S. hospitals: results of a 2002 survey. *J Am Med Inform Assoc* 2004;11:95–99.
9. Engle RL Jr. Attempts to use computers as diagnostic aids in medical decision making: a thirty-year experience. *Perspect Biol Med* 1992;35:207–219.
10. Berner ES, Webster GD, Shugerman AA, et al. Performance of four computer-based diagnostic systems. *N Engl J Med* 1994;330:1792–1796.
11. Miller R, Masarie FE, Myers J. Quick medical reference (QMR) for diagnostic assistance. *MD Comput* 1986;3:34–48.
12. Warner HR Jr. Iliad: moving medical decision-making into new frontiers. *Methods Inf Med* 1989;28:370–372.
13. Barnett GO, Cimino JJ, Hupp JA, Hoffer EP. DXplain—an evolving diagnostic decision-support system. *JAMA* 1987;258:67–74.
14. Heckerman DE, Horvitz EJ, Nathwani BN. Toward normative expert systems: part I. The Pathfinder project. *Methods Inf Med* 1992;31:90–105.
15. Shortliffe EH. The adolescence of AI in medicine: will the field come of age in the 90s? *Artif Intell Med* 1993;5:93–106.
16. Clayton PD, Hripcsak G. Decision support in healthcare. *Int J Biomed Comput* 1995;39:59–66.
17. Heathfield HA, Wyatt J. Philosophies for the design and development of clinical decision-support systems. *Methods Inf Med* 1993;32:1–8.

18. Buchanan BG, Feigenbaum EA. Dendral and Metadendral: their applications and dimension. *Artif Intell* 1978;11:5–24.
19. McDermott J. R1: A rule-based configurer of computer systems. *Artif Intell* 1982;19:39–88.
20. Russell S, Norvig P. *Artificial intelligence: a modern approach*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall; 2003.
21. Luger GF, Stubblefield WA. *Artificial intelligence and the design of expert systems*. Redwood City, CA: Benjamin/Cummings Publishing Company, Inc.; 1989.
22. Minsky M. A framework for representing knowledge. In: Haugeland J, ed. *Mind design*. Cambridge, MA: MIT Press; 1981:95–128.
23. Kokol P, Mernik M, Zavrsnik J, Kancler K, Malcic I. Decision trees based on automatic learning and their use in cardiology. *J Med Syst* 1994;18:201–206.
24. Baxt WG. Use of an artificial neural network for the diagnosis of acute myocardial infarction. *Ann Intern Med* 1991;115:843–848.
25. Pincioli F, Combi C, Pozzi G. Object oriented DBMS techniques for time oriented medical records. *Medinfo* 1992;17:231–241.
26. Frost RD, Gillenson ML. Integrated clinical decision support using an object-oriented database management system. *Methods Inf Med* 1993;32:154–160.
27. Long W. Medical diagnosis using a probabilistic causal network. *Appl Artif Intell* 1989;3:367–383.
28. Patil R, Szolovits P, Schwartz W. Causal understanding of patient illness and diagnosis. *Proc Seventh Int Joint Conf Artif Intell; Vancouver Canada* 1981: 893–899.
29. Allen JF. Towards a general theory of action and time. *Artif Intell Med* 1984;23:123–154.
30. Shahar Y, Musen M. A temporal-abstraction system for patient monitoring. *Proc Annu Symp Comput Appl Med Care* 1992:121–127.
31. Kahn MG, Fagan LM, Sheiner LB. Model-based interpretation of time-varying medical data. *Proc Annu Symp Comput Appl Med Care* 1989:28–32.
32. Kohane IS. Temporal reasoning in medical expert systems. Technical Report 389, Laboratory of Computer Science, Massachusetts Institute of Technology, Cambridge, MA, 1987.
33. Aliferis CF, Cooper GF, Miller RA, Buchanan BG, Bankowitz R, Guise N. A temporal analysis of QMR. *J Am Med Inform Assoc* 1996;3:79–91.
34. Reiter R. A logic for default reasoning. *Artif Intell* 1980;13:81–132.
35. Patil R. Review of causal reasoning in medical diagnosis. *Proc Annu Symp Comput Appl Med Care* 1986:11–16.
36. Miller RA. Internist-1/Caduceus: problems facing expert consultant programs. *Methods Inf Med* 1984;23:9–14.
37. Laursen P. Event detection on patient monitoring data using causal probabilistic networks. *Methods Inf Med* 1994;33:111–115.
38. Rutledge GW, Andersen SK, Polaschek JX, Fagan LM. A belief network model for interpretation of ICU data. *Proc Annu Symp Comput Appl Med Care* 1990:785–789.
39. Pradhan M, Provan G, Henrion M. Experimental analysis of large belief networks for medical diagnosis. *Proc Annu Symp Comput Appl Med Care* 1994: 775–779.

40. Pearl J. Probabilistic reasoning in intelligent systems: networks of plausible inference. San Mateo, CA: Morgan-Kaufman; 1988.
41. Sonnenberg FA, Hagerty CG, Kulikowski CA. An architecture for knowledge-based construction of decision models. *Med Decis Making* 1994;14:27–39.
42. Cooper G. Probabilistic and decision-theoretic systems in medicine. *Artif Intell Med* 1993;5:289–292.
43. Zadeh LA. Fuzzy logic, neural networks, and soft computing. *Commun ACM* 1994;37:77–84.
44. Zimmerman HJ. Fuzzy set theory and its applications, 2nd ed. Amsterdam: Kluwer-Nijhoff; 1990.
45. Maiers JE. Fuzzy set theory and medicine: the first twenty years. *Proc Annu Symp Comput Appl Med Care* 1985:325–329.
46. Shafer G. A mathematical theory of evidence. Princeton, NJ: Princeton University Press; 1976.
47. McDermott D, Doyle J. Non-monotonic logic-I. *Artif Intell* 1980;13:41–72.
48. Kolodner JL, Leake DB. A tutorial introduction to case-based reasoning. In: Leake D, ed. *Case-based reasoning: experiences, lessons and future directions*. Cambridge, MA: AAAI Press/MIT Press; 1996:31–67.
49. Stamper R, Todd BS, Macpherson P. Case-based explanation for medical diagnostic programs, with an example from gynaecology. *Methods Inf Med* 1994;33: 205–213.
50. Koton P. Reasoning about evidence in causal explanations. *Proc Seventh Nat Conf Artif Intell*, Vancouver, Canada, 1988:256–261.
51. Gruber, TR. A translation approach to portable ontologies. *Knowl Acquis* 1993;2:199–220.
52. Gruninger M, Lee J. Ontology applications and design. *Commun ACM* 2002;45:39–41.
53. Uschold M, Jasper R. A Framework for understanding and classifying ontology applications. *Proc IJCAI-99 Workshop on Ontologies and Problem-Solving Methods*, Intl Joint Conferences on Artificial Intelligence, Stockholm, Sweden, 1999.
54. Campbell KE, Oliver DE, Spackman KA, Shortliffe EH. Representing thoughts, words, and things in the UMLS. *J Am Med Inform Assoc* 1998;5: 421–431.
55. Goldberg H, Goldsmith D, Law V, Keck K, Tuttle M, Safran C. An evaluation of UMLS as a controlled terminology for the Problem List Toolkit. *Medinfo* 1998;9:609–612.
56. McCray AT, Nelson SJ. The representation of meaning in the UMLS. *Methods Inf Med* 1995;34:193–201.
57. Burgun A, Bodenreider O. Mapping the UMLS Semantic Network into general ontologies. *Proc AMIA Symp* 2001:81–85.
58. Rector AL, Nowlan WA. The GALEN project. *Comput Methods Programs Biomed* 1994;45:75–78.
59. Rector AL, Rogers JE. Ontological issues in using description logic to represent medical concepts: experience from GALEN. IMIA, Phoenix, AZ; 1999 (also available at: www.opengalen.org/resources.html).
60. Rector AL, Bechhofer S, Goble CA, Horrocks I, Nowlan WA, Solomon WD. The GRAIL concept modelling language for medical terminology. *Artif Intell Med* 1997;9:139–171.

61. Wang AY, Barrett JW, Bentley T, et al. Mapping between SNOMED RT and Clinical Terms version 3: a key component of the SNOMED CT development process. *Proc AMIA Symp* 2001:741–745.
62. Spackman KA, Campbell KE, Cote RA. SNOMED RT: a reference terminology for health care. *Proc AMIA Annu Fall Symp* 1997:640–644.
63. Chisolm J. The read classification. *Br Med J* 1990;300:1092–1096.
64. Spackman KA. Normal forms for description logic expressions of clinical concepts in SNOMED RT. *Proc AMIA Symp* 2001:627–631.
65. Bakken S, Warren JJ, Lundberg C, et al. An evaluation of the usefulness of two terminology models for integrating nursing diagnosis concepts into SNOMED Clinical Terms. *Int J Med Inform* 2002;68:71–77.
66. Spackman KA. Integrating sources for a clinical reference terminology: experience linking SNOMED to LOINC and drug vocabularies. *Medinfo* 1998;9: 600–603.
67. Denny M. Ontology building: a survey of editing tools. (Accessed December 17, 2003 at <http://www.xml.com/pub/a/2002/11/06/ontologies.html>.)
68. Noy NF, McGuinness DL. Ontology development 101: a guide to creating your first ontology. (Accessed July 17, 2006 at http://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html.)
69. Holsapple CW, Joshi KD. A Collaborative approach to ontology design. *Commun ACM* 2002;45:42–47.
70. Guarino N, Welty C. Evaluating ontological decisions with ontoclean. *Commun ACM* 2002;45:61–65.
71. Dick RS, Steen EB, eds. *The computer-based patient record: an essential technology for health care*. National Academy Press, Washington, DC; 1992.
72. Hammond WE. Call for a standard clinical vocabulary. *J Am Med Inform Assoc* 1997;4:254–255.
73. Rose JS, Fisch BJ, Hogan WR, et al. Common medical terminology comes of age, part two: current code and terminology sets—strengths and weaknesses. *J Healthc Inf Manage* 2001;15:319–330.
74. Liu GC, Cooper JG, Schoeffler KM, Hammond WE. Standards for the electronic health record, emerging from health care’s Tower of Babel. *Proc AMIA Symp* 2001:388–392.
75. Peden AH. An overview of coding and its relationship to standardized clinical terminology. *Top Health Inf Manage* 2000;21:1–9.
76. Fenton SH. Clinical vocabularies and terminologies: impact on the future of health information management. *Top Health Inf Manage* 2000;21:74–80.
77. Chute CG, Cohn SP, Campbell JR. A framework for comprehensive health terminology systems in the United States: development guidelines, criteria for selection, and public policy implications. ANSI Healthcare Informatics Standards Board Vocabulary Working Group and the Computer-Based Patient Records Institute Working Group on Codes and Structures. *J Am Med Inform Assoc* 1998;5:503–510.
78. Elkin PL, Brown SH, Carter J, et al. Guideline and quality indicators for development, purchase and use of controlled health vocabularies. *Int J Med Inf* 2002;68:175–186.
79. Van Bommel JH, Musen MA, eds. *Handbook of medical informatics*. Heidelberg: Springer-Verlag; 1997.

80. Gershenov M. The ICD family of classifications. *Methods Inf Med*. 1995; 34:172–175.
81. International Statistical Classification of Diseases and Related Health Problems, 10th revision. Geneva: World Health Organization; 1993.
82. Stausberg J, Lang H, Obertacke U, Rauhut F. Classifications in routine use: lessons from ICD-9 and ICPM in surgical practice. *J Am Med Inform Assoc*. 200;8:92–100.
83. Movig KL, Leufkens HG, Lenderink AW, Egberts AC. Validity of hospital discharge international classification of diseases (ICD) codes for identifying patients with hyponatremia. *J Clin Epidemiol* 2003;56:530–535.
84. Geraci JM, Ashton CM, Kuykendall DH, Johnson ML, Wu L. International classification of diseases, 9th revision, clinical modification codes in discharge abstracts are poor measures of complication occurrence in medical inpatients. *Med Care* 1997;35:589–602.
85. Current Procedural Terminology 2000. Chicago: American Medical Association; 2000.
86. Huff SM, Rocha RA, McDonald CJ, et al. Development of the logical observation identifier names and codes (LOINC) vocabulary. *J Am Med Inform Assoc* 1998;5:276–292.
87. McDonald CJ, Huff SM, Suico JG, et al. LOINC, a universal standard for identifying laboratory observations: a 5-year update. *Clin Chem* 2003;49: 624–633.
88. Forrey AW, McDonald CJ, DeMoor G, et al. Logical observation identifier names and codes (LOINC) database: a public use set of codes and names for electronic reporting of clinical laboratory test results. *Clin Chem* 1996;42: 81–90.
89. Cimino JJ. Desiderata for controlled medical vocabularies in the twenty-first century. *Methods Inf Med* 1998;37:394–403.
90. Rector AL. Clinical terminology: why is it so hard? *Methods Inf Med* 1999;38:239–252.
91. van Ginneken AM, Verkoijen MJ. A multi-disciplinary approach to a user interface for structured data entry. *Medinfo* 2001;10:693–697.
92. Warren JJ, Collins J, Sorrentino C, Campbell JR. Just-in-time coding of the problem list in a clinical environment. *Proc AMIA Symp* 1998:280–284.
93. Elkin PL, Tuttle M, Keck K, Campbell K, Atkin G, Chute CG. The role of compositionality in standardized problem list generation. *Medinfo* 1998;9: 660–664.
94. McKnight LK, Elkin PL, Ogren PV, Chute CG. Barriers to the clinical implementation of compositionality. *Proc AMIA Symp* 1999:320–324.
95. Rector AL, Solomon WD, Nowlan WA, Rush TW, Zanstra PE, Claassen WM. A terminology server for medical language and medical information systems. *Methods Inf Med* 1995;34:147–157.
96. Chute CG, Elkin PL, Sherertz DD, Tuttle MS. Desiderata for a clinical terminology server. *Proc AMIA Symp* 1999:42–46.
97. Dolin RH, Spackman KA, Markwell D. Selective retrieval of pre- and post-coordinated SNOMED concepts. *Proc AMIA Symp*. 2002:210–214.
98. Spackman KA. Normal forms for description logic expressions of clinical concepts in SNOMED RT. *Proc AMIA Symp* 2001:627–631.

99. Weed LL. Knowledge coupling: new premises and new tools for medical care and education. New York: Springer-Verlag; 1991.
100. Carter JH, ed. Electronic medical records: a guide for clinicians and administrators. Philadelphia: American College of Physicians; 2001.
101. Kuperman GJ, Sittig DF, Shabot MM. Clinical decision support for hospital and critical care. *J Healthc Inf Manag* 1999;13:81–96.
102. Overage JM, Tierney WM, McDonald CJ. Clinical decision support in ambulatory care: tools, trials, tribulations. *J Healthc Inf Manag* 1999;13:67–79.
103. Aranow M. What works: clinical information systems. Order entry rules. Healthcare enterprise achieves physician acceptance, reduced medication errors and improved patient outcomes through CIS and CPOE technology. *Health Manag Technol* 2002;7:34, 38.
104. Murray MD, Harris LE, Overhage JM, et al. Failure of computerized treatment suggestions to improve health outcomes of outpatients with uncomplicated hypertension: results of a randomized controlled trial. *Pharmacotherapy* 2004;3:324–337.
105. Subramanian U, Fihn SD, Weinberger M, et al. A controlled trial of including symptom data in computer-based care suggestions for managing patients with chronic heart failure. *Am J Med* 2004;6:375–384.
106. Tierney WM, Overhage JM, Murray MD, et al. Effects of computerized guidelines for managing heart disease in primary care. *J Gen Intern Med* 2003;18:967–976.

5 Diagnostic Decision Support Systems*

RANDOLPH A. MILLER and ANTOINE GEISSBUHLER

Since primeval times, mankind has attempted to explain natural phenomena using models. For the past four decades, a new kind of modeler, the healthcare informatician, has developed and proliferated a new kind of model, the Clinical Diagnostic Decision Support System (DDSS). Modeling historically was, and still remains, an inexact science. Ptolemy, in the “Almagest,” placed the earth at the center of the universe and could still explain why the sun would rise in the east each morning. Newton’s nonrelativistic formulation of the laws of mechanics works well for earth-bound engineering applications. Past and present DDSS incorporate inexact models of the incompletely understood and exceptionally complex process of clinical diagnosis. Yet mankind, using imperfect models, has built machines that fly, and has cured many diseases. Because DDSS augment the natural capabilities of human diagnosticians, they have the potential to be employed productively.¹

This chapter presents a definition of clinical diagnosis and of DDSS; a discussion of how humans accomplish diagnosis; a survey of previous attempts to develop computer-based clinical diagnostic tools; a discussion of the problems encountered in developing, implementing, evaluating, and maintaining clinical diagnostic decision support systems; and a brief discussion of the future of such systems.

Definitions of Diagnosis

In order to understand the history of clinical diagnostic decision support systems and envision their future roles, it is important to define clinical

* Portions of this chapter have been taken verbatim, with permission of the American Medical Informatics Association (AMIA), which owns the copyrights, from: Miller RA, Medical Diagnostic Decision Support Systems—Past, Present, and Future: A Threaded Bibliography and Commentary. *J Am Med Inform Assoc* 1994;1:8–27; and from Miller RA, Evaluating Evaluations of Medical Diagnostic Systems. *J Am Medical Inform Assoc* 1996;3:429–431.

diagnosis and computer-assisted clinical diagnosis. A simple definition of diagnosis is:²

the placing of an interpretive, higher level label on a set of raw, more primitive observations [Definition 1].

By this definition one form of diagnosis might consist of labeling, as “abnormal” any laboratory test results falling outside 1.5 times the 95% confidence intervals for the “normal” values seen in the general population as measured by that laboratory. Another level of diagnosis under the same definition might consist of labeling the combination of a low serum bicarbonate level, a high serum chloride level, and an arterial blood pH of 7.3 as “metabolic acidosis.”

A more involved definition of diagnosis, specific for clinical diagnosis, is:²

a mapping from a patient's data (normal and abnormal history, physical examination, and laboratory data) to a nosology of disease states [Definition 2].

Both of these definitions treat diagnosis improperly as a single event, rather than as a process. A more accurate definition is found in the Random House Collegiate Dictionary. There, diagnosis is defined as:³

“the process of determining by examination the nature and circumstances of a diseased condition” [Definition 3].

Skilled diagnosticians develop an understanding of what the patient's life situation was like before the illness began, how the illness has manifested itself, and how it has affected the life situation.² The clinician must also determine the patient's understanding of, and response to, an illness. The process of diagnosis entails a sequence of interdependent, often highly individualized tasks: evoking the patient's initial history and physical examination findings; integration of the data into plausible scenarios regarding known disease processes; evaluating and refining diagnostic hypotheses through selective elicitation of additional patient information, such as laboratory tests or serial examinations; initiating therapy at appropriate points in time (including before a diagnosis is established); and evaluating the effect of both the illness and the therapy, on the patient, over time.²

Diagnosis is a process composed of individual steps. These steps go from a point of origin (a question and a set of “presenting findings” and “previously established diagnoses”), to a point of destination (an answer, usually consisting of a set of “new established diagnoses” and/or “unresolved differential diagnoses”). While the beginning and end points may be identical, the steps one diagnostician follows may be very different from those taken by another diagnostician, and the same diagnostician may take different steps in two nearly identical cases. Because expertise varies among clini-

cians, different individuals will encounter different diagnostic problems in evaluating the same patient. For instance, they may generate dissimilar questions based on difficulties with disparate steps in the diagnostic process, even if they follow exactly the same steps.

Studies of clinicians' information needs help us to understand the variability in diagnostic problem solving among clinicians. Osheroff, Forsythe, and colleagues^{4,5} used participant observation, a standard anthropological technique, to identify and classify information needs during the practice of medicine in an academic health center. They identified three components of "comprehensive information needs:" (1) currently satisfied information needs (information recognized as relevant to a question and already known to the clinician); (2) consciously recognized information needs (information recognized by the clinician as important to know to solve the problem, but which is not known by the clinician); and (3) unrecognized information needs (information that is important for the clinician to know to solve a problem at hand, but is not recognized as being important by the clinician). Failure to detect a diagnostic problem at all would fall into the latter category. Different clinicians will experience different diagnostic problems within the same patient case, based on each clinician's varying knowledge of the patient and unique personal store of general medical knowledge. Osheroff et al. noted the difficulty people and machines have in tailoring general medical knowledge to specific clinical cases. There may be a wealth of information in a patient's inpatient and outpatient records, and also a large medical literature describing causes of the patient's problems. The challenge is to quickly and efficiently reconcile one body of information with the other.^{1,4} DDSS can potentially facilitate that reconciliation. A DDSS can be defined as:

a computer-based algorithm that assists a clinician with one or more component steps of the diagnostic process [Definition 4].

While clinicians may have differing conceptions of what they mean by diagnosis, the definitions embodied in DDSS are even more varied. DDSS users are often slow to recognize that each system functionally defines diagnosis as the set of tasks that it can perform. Experienced users often become familiar with using DDSS as tools to supplement, rather than replace, their own diagnostic capabilities. Untrained DDSS users may have preconceived unrealistic expectations that engender subsequent frustration. Naive users view diagnosis on their own terms, based on their own experiences, and expect diagnostic decision support systems to behave in a familiar manner. For example, it is unreasonable to expect that a DDSS can solve a vague problem with minimal input or that DDSS can assist clinicians in understanding how an illness has affected the patient's lifestyle. Conversely, system developers sometimes create useful diagnostic tools that provide capabilities outside the experience of human diagnosticians. For

example, the relationships function of R-QMR[†] (a DDSS), takes, as input, up to 10 findings that the clinician-user would like to explain as the key or “pivotal” findings from a diagnostically challenging case, and produces, as output, a rank-ordered list of “disease complexes” that each explain all of the input findings.⁷ Each disease complex is made up of from one to four interrelated disorders (e.g., disease A predisposing to disease B and causing disease C). Because busy clinicians can spare little free time for extraneous activities, user training for DDSS is extremely critical and must address the potential cognitive mismatch between user expectations and system capabilities.

An important concept related to the use of DDSS is understanding that the problem to be solved originates in the mind of the clinician-user. The diagnostic problem cannot be defined in an absolute sense, for example, by an arbitrary set of input findings selected from a case. The DDSS analog of the metaphysical question, “if a tree falls in a forest in the absence of people, will there be a sound?” is “if clinical findings are extracted from a patient case in the absence of a query from a clinician caring for the patient (or someone asked to function with that mindset), is there a diagnostic problem to be solved, or can there be a ‘correct’ answer?” There is only one way that the findings of a case, in isolation, can define a diagnostic problem, and that is when the diagnostic problem is the global one, i.e., the DDSS, through its own initiative, is expected to take all the steps in the diagnostic process required to explain all patient findings through establishing new diagnoses (or unresolved differential diagnoses if there is not a solution). It is rare in clinical practice to encounter the “global” diagnostic problem. Clinicians usually complete a portion of the evaluation process before they encounter difficulty, and, correspondingly, once they overcome the difficulty, they are usually capable of completing the evaluation without further assistance. While early DDSS developers often assumed the only problem worth solving was the global diagnostic problem, emphasis over the last decade has shifted to helping clinicians with problems they encounter during individual steps in the diagnostic process. This has led to the demise of the “Greek Oracle” model, where the DDSS was expected to take all of the patient’s findings and come up with “the answer.” Current DDSS models assume that the user will interact with the DDSS in an iterative fashion, selectively entering patient information and using the DDSS output to assist with the problems encountered in the diagnostic process.⁸

In order to interact optimally with the DDSS, the users need to understand the assumptions built into the system. As noted previously, each DDSS functionally defines diagnosis as the tasks it can perform (or assist

[†] In this chapter, R-QMR refers to the noncommercial, research version of QMR, the DDSS developed by Miller, Masarie, and Myers.⁶ The commercial version of QMR, previously marketed by FirstDataBank, while initially identical to R-QMR in 1990, was developed independently of R-QMR since that time.

users in performing). The subtle nature of underlying assumptions that system developers incorporate into DDSS can be deceptive to users. For example, one of the most well known diagnostic systems is the Bayesian program for diagnosis of acute abdominal pain developed by de Dombal and colleagues.^{9,10} As it was originally developed, the system's goal, not stated explicitly, was to discriminate between surgical and nonsurgical causes of acute abdominal pain in order to help triage patients in an emergency room (or similar) setting. A limited number of explicit diagnoses are supported by the system, all of which, except "nonspecific abdominal pain," are surgical disorders or potentially surgically treated disorders (such as acute appendicitis, acute pancreatitis, and acute diverticulitis). The performance of the system was evaluated in multicenter studies¹⁰ and shown to be exemplary with respect to the circumstances for which it was designed. However, naive users generically relying on de Dombal's system to help with the diagnosis of all patients presenting with acute abdominal pain would be disappointed. There is a high potential for errors in caring for such patients if the clinician-users do not supplement system output with their own knowledge. The system could not properly diagnose patients presenting with acute intermittent porphyria, lead poisoning, early T10 dermatome herpes zoster, or familial Mediterranean fever. Even when the system performs optimally, all these conditions would be labeled as "nonspecific abdominal pain."

The utility of making specific diagnoses lies in the selection of effective therapies, making accurate prognoses, and providing detailed explanations.¹ In some situations, it is not necessary to arrive at an exact diagnosis in order to fulfill one or more of these objectives. Treatment is often initiated before an exact diagnosis is made. Furthermore, the utility of making certain diagnoses is debatable, especially if there is a small probability of effective treatment. For instance, labeling a patient as having "obesity" does not flatter the patient, and, even worse, may cause the clinician to do more harm than good. Good documentation exists in the medical literature, that once a patient reaches approximately twice their ideal body weight, the patient's metabolism and psychology related to eating changes¹¹ so that the prognosis of dieting down to the ideal body weight and staying there is approximately equal to the survival rate for gastric carcinoma at five years. Resorting to nonprescription, potentially harmful therapies such as unsupervised prolonged fasting, or prescribed amphetamines may lead to more harm than benefit, yet desperate patients and physicians sometimes resort to such approaches.

The cost of eliciting all possible patient data is potentially staggering—temporally, economically, and ethically—since there are real risks of morbidity and/or mortality associated with many diagnostic procedures such as liver biopsy or cardiac catheterization. (Of note, there are some individuals who now advocate "total body imaging" as a noninvasive, relatively affordable mechanism of gathering maximal diagnostic information at one

time from a patient, despite the fact that, other than for identifying tumor metastases, there is little evidence to support this procedure.) Given the impossibility and impracticality of gathering every conceivable piece of diagnostic information with respect to each patient, the “art” of diagnosis lies in the ability of the diagnostician to carefully evoke enough relevant information to justify all important and ultimately correct diagnoses in each case, as well as to initiate therapies at appropriate points during the evaluation.² The knowledge of how to “work up” the patient depends critically on the ability to evoke history, symptoms, and physical examination findings, concurrently with the ability to generate diagnostic hypotheses that suggest how to further refine or pursue the findings already elicited, or to pursue completely different additional findings. In addition, this must be done in a compassionate and cost-effective manner.²

Human Diagnostic Reasoning

Diagnostic reasoning involves diverse cognitive activities, including information gathering, pattern recognition, problem solving, decision making, judgment under uncertainty, and empathy. Large amounts of highly organized knowledge are necessary to function in this relatively unstructured cognitive domain. Our knowledge of human diagnostic reasoning is based on generic psychological experiments about reasoning and on direct studies of the diagnostic process itself. Relevant principles of human problem-solving behavior have been unveiled through focused studies examining constrained problem spaces such as chess-playing and cryptoarithmetic.¹² Such studies have documented that experts recognize patterns of activity within a domain at an integrated, higher level (“chunking”) than novices. Additional psychological experiments about judgments made under uncertainty¹³ have provided insights into individuals’ imperfect semiquantitative reasoning skills.

To investigate the complex intellectual task of clinical diagnosis, many researchers^{14,15} have used behavioral methods that combine protocol analysis with introspection. Researchers record clinicians as they think aloud while performing specified cognitive tasks related to diagnosis (including normal clinical activities). Post facto, the clinicians themselves, or others, are asked to interpret the motives, knowledge, diagnostic hypotheses, and strategies involved in the recorded sessions. However, there is no proof that the stories constructed by experts to explain their diagnostic reasoning correspond to the actual reasoning methods they use subconsciously.

Most models of diagnostic reasoning include the following elements: the activation of working hypotheses; the testing of these hypotheses; the acquisition and interpretation of additional information; and confirming, rejecting, or adding of new hypotheses as information is gathered over time. Working hypotheses are generated early in the process of information

gathering, at a time when only few facts are known about the patient.^{14,15} Only a limited number of these hypotheses, rarely more than five, are entertained simultaneously, probably because of the limited capacity of human short term memory.¹⁶ Early hypothesis generation is probably accomplished through some form of pattern recognition, with experts more capable of applying compiled knowledge and experiences than novices. Comparing clinical reasoning in novices and experts, Evans and Patel¹⁷ showed that experts rarely rely directly on causal reasoning and knowledge of basic sciences, except when reasoning outside their domain of expertise.

As noted by Pople and others,¹⁸ clinical diagnosis fits Simon's criteria for being an ill-structured problem.¹⁹ Simon gave as an example of an ill-structured problem, the task an architect faces in creatively designing a new house "from scratch"—the realm of possible solutions encompasses a great variety of applicable methods and a broad set of alternative outcomes. As summarized by Pople, Simon observed that ill-structured problems can be solved by splitting the problem into smaller, well defined subtasks that are each more easily accomplished.¹⁸

In clinical diagnosis, early hypothesis generation helps to constrain reasoning to "high yield" areas, and permits the use of heuristic methods to further elucidate a solution.²⁰ Studies have shown that most clinicians employ the hypothetico-deductive method after early hypothesis generation.^{14,15} Data are collected with a view to their usefulness in refining, rejecting, or substituting for the original set of hypotheses. In the setting of clinicopathological exercises, Eddy and Clanton²¹ showed that identification of a pivotal finding is often used to simplify the diagnostic problem and to narrow the focus to a limited set of hypotheses. Kassirer and Gorry¹⁵ described the "process of case building," where hypotheses are evaluated against the model of a disease entity using techniques that can be emulated in computers using Bayes' rule, Boolean algebra, or template matching (see Chapter 2 for an explanation of these terms). They also recognized that heuristic methods are commonly used to confirm, eliminate, discriminate between, or explore hypotheses. Weed²² and Hurst and Walker²³ suggested that clinical problem-solving can be approached by splitting complex, composite problems into relatively independent, discrete "problem areas." With respect to diagnosis, Pople observed that separating complex differential diagnoses into problem areas allows diagnosticians to apply additional powerful reasoning heuristics. They can assume that the differential diagnosis list within a problem area contains mutually exclusive hypotheses and that the list can be made to be exhaustive (i.e., complete), so that it is assured that the correct diagnosis is on the list for the problem area, and that only one diagnosis on the list is the correct one.¹⁸

Kassirer has identified three abstract categories of human diagnostic reasoning strategies: probabilistic, causal, and deterministic.²⁴ Formal models for each type of reasoning have been developed, most often separately from observational studies on how actual reasoning occurs. Probabilistic models

such as Brunswik's lens model²⁵ and Bayesian^{26,27} approaches, as well as decision analysis^{28,29} define statistical associations between clinical variables and use mathematical models to compute optimal decisions. While it is clear that diagnosticians consider prevalence and other probabilistic concepts during their reasoning,^{14,15} observational and experimental studies show that humans are not intuitively good statisticians.^{13,30} Human problem-solvers tend to rely on judgmental heuristics. Experiments document that humans improperly evaluate subjective probabilities, misuse prior probabilities, and fail to recognize important phenomena, such as the regression towards the mean. While there has been some evidence that humans have more difficulty reasoning with probabilities than they do understanding the concepts which underlie them,³¹ humans also demonstrate other reasoning errors such as reluctance to revise opinions when presented with data that do not fit with working hypotheses when the data's diagnostic significance is properly understood.^{13,30}

Models of causal (pathophysiological) reasoning, such as those developed by Feinstein^{32,33} in the 1970s, establish cause-and-effect relations between clinical variables within anatomic, physiologic, biochemical, and ultimately, genetics-based representations of the reality. Although causal inferences (reasoning from causes to consequences) can be viewed as the inverse of diagnostic inferences (reasoning from consequences to causes), studies have shown that when making judgments under uncertainty, humans assign greater impact to causal rather than other diagnostic data of equal informative weight, and they commonly make over-confident predictions when dealing with highly uncertain models.¹³ Causal, pathophysiological reasoning uses shared, global, patient-independent knowledge³³ and provides an efficient means of verifying and explaining diagnostic hypotheses. However, it is not clear how much causal reasoning is actually used in early hypothesis generation and other stages of nonverbalized diagnostic reasoning. As noted earlier, observational studies indicate that experts tend to employ causal, pathophysiological reasoning only when faced with problems outside the realm of their expertise, highly atypical problems, or when they are asked to explain their reasoning to others.⁵

In deterministic models, production rules, i.e., specifying appropriate actions in response to certain conditions, are used to represent the basic building blocks of human problem-solving. Such if-then rules representing compiled knowledge can be expressed in the form of branching-logic flowcharts and clinical algorithms for nonexperts to follow. However, production rules do not deal effectively with uncertainty,³⁴ which is a disadvantage in clinical practice, where uncertainty is a common feature.

The late M. Scott Blois, a great philosopher-informatician-clinician, used a funnel to illustrate the spectrum of clinical judgment.³⁵ Consideration of patients' ill-structured problems, including undifferentiated concerns and vague complaints, occurs at the wide end of the funnel. Focused decisions in response to specific clinical questions (e.g., choosing an antibiotic to treat

the bacteria isolated as the cause of a pneumonia) were represented at the narrow end. This model is consistent with Simon's view of how humans solve ill-structured problems.¹⁸ Blois noted that decision support systems were best applied toward the narrow end of the funnel, since circumscribed, well structured problems are encountered there. Those problems are more amenable to solution through application of computational models of cognitive skills, requiring only focused and specific knowledge. On the other hand, at the open end of the funnel, one has to deal with common-sense knowledge and the general scope of ordinary human judgment in order to make meaningful progress, and few computer-based systems (other than those for record-keeping) are applicable.

Historical Survey of Diagnostic Decision Support Systems

The majority of important concepts related to current DDSS were developed and presented in the literature prior to 1976. In a comprehensive 1979 review of reasoning strategies employed by early DDSS, Shortliffe, Buchanan, and Feigenbaum identified the following classes of DDSS: clinical algorithms, clinical databanks that include analytical functions, mathematical pathophysiological models, pattern recognition systems, Bayesian statistical systems, decision-analytical systems, and symbolic reasoning (sometimes called "expert" systems).³⁶ This section, without being comprehensive, will describe how some of the early pioneering efforts led to many classes of systems present today.

The many types of DDSS correspond to the large number of clinical domains to which diagnostic reasoning can be applied, to the multiple steps of diagnostic reasoning described above and to the variety of difficulties that diagnosticians may encounter at each step. When health care informatics researchers come upon the term "clinical diagnostic decision-support systems," many think primarily of general-purpose, broad-spectrum consultation systems.¹ However, Definitions 1 through 3, in the section on definitions of diagnosis, form the basis for the broad spectrum of diagnostic systems actually encountered. In a sense, Definition 1, diagnosis as interpretation of raw observations, is potentially recursive, as it defines successively more complex classes of diagnostic tools. Low-level diagnostic labels placed on "raw" observations can be used as input into second-level diagnostic systems that produce higher-level labels that are then used at progressively higher levels.

There are systems for general diagnosis (no matter how broad or narrow their application domains), and systems for diagnosis in specialized domains such as interpretation of ECG tracings.³⁷ The general notion of DDSS conveyed in the biomedical literature sometimes overlooks specialized, focused, yet highly successful medical device-associated diagnostic systems. Some simple DDSS help to interpret blood gas results, assist in

categorizing diagnostic possibilities based on the output of serum protein electrophoresis devices, or aid in the interpretation of standardized pulmonary function tests. DDSS for cytological recognition and classification have found successful application in devices such as automated differential blood count analyzers and systems to analyze Papanicolaou smears.¹ Small, focused DDSS are the most widely used form of diagnostic decision support programs, and their use will grow as they are coupled with other automated medical devices.¹

In their classic paper published in 1959, Ledley and Lusted²⁶ observed that physicians have an imperfect knowledge of how they solve diagnostic problems. Ledley and Lusted detailed the principles underlying work on Bayesian and decision-analytic diagnostic systems that has been carried out over subsequent decades. They stated that both logic (as embodied in set theory and Boolean algebra) and probabilistic reasoning (as embodied in Bayes' rule) were essential components of medical reasoning. Ledley and Lusted mentioned the importance of protocol analysis in understanding human diagnostic reasoning. They stated that they had reviewed how physicians solve *New England Journal of Medicine* CPC (clinicopathological conference) cases as the foundation for their work on diagnostic computer systems. Both for practical reasons and for philosophical reasons, much work on DDSS has focused on the differences between logical deductive systems and probabilistic systems. Chapter 2 describes these approaches in more detail. What follows is a description of how DDSS have embodied these reasoning principles.

Logical systems, based on "discriminating questions" to distinguish among mutually exclusive alternatives, have played an important role since the pioneering work by Bleich and his colleagues³⁸ on acid base and electrolyte disorders. To this day, such systems are applicable to narrow domains, especially those where it is fairly certain that only one disorder is present. When users of a branching logic system incorrectly answer one of the questions posed by the system, they may find themselves "out on a limb" with no way to recover except by starting over from the beginning; the likelihood of such problems increases when multiple independent disease processes interact in the patient. Thus, ideal application areas are those where detailed knowledge of pathophysiology or extensive epidemiological data make it possible to identify parameters useful for dividing diagnostic sets into nonintersecting subsets, based on specific characteristics.

Bayes' rule is applicable to larger domains. Warner and colleagues in 1960–1961 developed one of the first medical application systems based on Bayes' rule. In their original contribution,²⁷ they discussed the independence assumption required among diagnoses and among findings by the most commonly employed Bayesian applications, and proposed a method for eliminating the influence of redundant findings. They obtained the probabilities used in the diagnosis of congenital heart diseases from literature review, from their own series of over 1,000 cases, and from experts' esti-

mates based on knowledge of pathophysiology. Warner et al. observed how diagnostic systems can be very sensitive to false positive findings and to errors in the system's database. The importance of obtaining accurate data from the user was emphasized. In their evaluation of their system's performance, they pointed out the need for an independent "gold standard" against which the performance of the system could be judged. In the evaluation of their system, they used cardiac catheterization data and/or anatomical (postmortem) data to confirm the actual patient diagnoses. Warner et al. have continued to develop and refine models for Bayesian diagnosis over the years.¹

In 1968, Gorry and Barnett developed a model for sequential Bayesian diagnosis.³⁹ The first practical Bayesian system, and one of the first DDSS to be utilized at widespread clinical sites, was the system for diagnosis of acute abdominal pain developed by de Dombal and colleagues.^{1,9} A large number of groups have subsequently developed, implemented, and refined Bayesian methods for diagnostic decision making, and a wave of enthusiasm surrounds current work on Bayesian belief networks for clinical diagnosis.¹ Probabilistic systems have played, and will continue to play, an important role in DDSS development.

An additional alternative exists to categorical (predicate calculus)⁴⁰ and probabilistic reasoning that combines features of both but retains a fundamental difference. That alternative is heuristic reasoning, reasoning based on empirical rules of thumb. The HEME program for diagnosis of hematological disorders was one of the earliest systems to employ heuristics and also one of the first systems to use, in effect, criteria tables for diagnosis of disease states. It was developed initially by Lipkin, Hardy, Engle, and their colleagues in the late 1950s.^{1,41-43} Programs which heuristically match terminology from stored descriptions of disease states to lexical descriptions of patient cases are similar conceptually to HEME. The CONSIDER program developed by Lindberg et al.⁴⁴ and the RECONSIDER program developed by Blois and his colleagues⁴⁵ used heuristic lexical matching techniques to identify diseases in Current Medical Information and Terminology (CMIT), a manual of diseases compiled and previously maintained by the American Medical Association. The EXPERT system shell, developed by Weiss and Kulikowski,⁴⁶ has been used extensively in developing systems that utilize criteria tables, including AI/Rheum,^{47,48} for diagnosis of rheumatological disorders, as well as other systems.

G. Anthony Gorry was an enlightened pioneer in the development of heuristic diagnostic systems that employ symbolic reasoning. In a classic paper published in 1968, Gorry⁴⁹ outlined the general principles underlying expert system approaches to medical diagnosis that were subsequently developed in the 1970s and 1980s. Gorry proposed a formal definition of the diagnostic problem. In a visionary manner, he analyzed the relationships among a generic inference function (used to generate diagnoses from observed findings), a generic test-selection function that dynamically selects

the best test to order (in terms of cost and information content), and a pattern-sorting function that is capable of determining if competing diagnoses are members of the same “problem area” (i.e., whether diagnostic hypotheses should be considered together because they are related to pathology in the same organ system). He pointed out the difference between the information value, the economic cost, and the morbidity or mortality risk of performing tests; discussed the cost of misdiagnosis of serious, life-threatening or disabling disorders; noted the potential influence of “red herring” findings on diagnostic systems; described the “multiple diagnosis” problem faced by systems when patients have more than one disease; and suggested that the knowledge bases underlying diagnostic systems could be used to generate simulated cases to test the diagnostic systems.

Gorry’s schemata represent the intellectual ancestors of a diverse group of medical diagnostic systems, including, among others: PIP (the Present Illness Program), developed by Pauker et al.; MEDITEL for adult illnesses, which was developed by Waxman and Worley from an earlier pediatric version; Internist-I developed by Pople, Myers, and Miller; QMR, developed by Miller, Masarie, and Myers; DXplain, developed by Barnett and colleagues; Iliad, developed by Warner and colleagues; and a large number of other systems.^{1,50-56}

Shortliffe introduced the clinical application of rule-based expert systems for diagnosis and therapy through his development of MYCIN^{1,57} in 1973–1976. MYCIN used backward chaining through its rule base to collect information to identify the organism(s) causing bacteremia or meningitis in patients (see discussion of backward and forward chaining in Chapter 2). A large number of rule-based DDSS have been developed over the years, but most rule-based DDSS have been devoted to narrow application areas due to the extreme complexity of maintaining rule-based systems with more than a few thousand rules.¹

With the advent of the microcomputer came a change in philosophy in regard to the development of DDSS. For example, the style of diagnostic consultation in the original 1974 Internist-I program treated the physician as unable to solve a diagnostic problem. The model assumed that the physician would transfer all historical information, physical examination findings, and laboratory data to the Internist-I expert diagnostic consultant program. The physician’s subsequent role was that of a passive observer, answering yes or no to questions generated by Internist-I. Ultimately, the omniscient Greek Oracle (consultant program) was supposed to provide the correct diagnosis and explain its reasoning. By the late 1980s and early 1990s, developers abandoned the “Greek Oracle” mode¹⁸ of diagnostic decision support. Encouraged by the critiquing model developed by Miller^{1,58} and his colleagues, recent DDSS developers have made it their objective to create a mutually beneficial system that takes advantage of the strengths of both the user’s knowledge and the system’s abilities. The goal is to improve

performance of both the user and the machine over their native (unassisted) states.

Several innovative techniques were added in the 1980s and 1990s, to previous models for computer-assisted medical diagnosis. The trend has been to develop more formal models that add mathematical rigor to the successful, but more arbitrary, heuristic explorations of the 1970s and early 1980s. However, there are tradeoffs involved in formal mathematical models, often related to available data quality, which in many ways make them heuristic as well.⁵⁹ Systems based on fuzzy set theory and Bayesian belief networks were developed to overcome limitations of heuristic and simple Bayesian models.¹ Reggia, Nau, and Wang^{1,60} developed set covering models as a formalization of ad hoc problem-area formation (partitioning) schemes, such as that developed by Pople and Miller for Internist-I.⁶¹

Neural networks represent an entirely new approach to medical diagnosis, although the weights learned by simple one-layer networks may be analogous or identical to Bayesian probabilities.¹ Problems with neural networks include selecting the best topology, preventing overtraining and undertraining, and determining what cases to use for training. The more complex a neural network is (number of input and output nodes, number of hidden layers), the greater the need for a large number of appropriate training cases. Often, large epidemiologically controlled patient data sets are not available. There is a tendency among some developers to resort to simulation techniques to generate training cases. Use of “artificial” cases to train neural networks may lead to sub-optimal performance on real cases. Chapters 1–4 provide additional detail on the models mentioned above.

Developing, Implementing, Evaluating, and Maintaining Diagnostic Decision Support Systems

For any DDSS to achieve success, it must complete a number of stages of development.^{2,62} To begin with, a DDSS should be developed to meet documented information needs.^{4,5,63} Developers must perform a clinical needs assessment to determine the utility of the proposed system and the frequency with which it might be used in various real-world settings. Clinical systems should not be developed simply because someone wants to test an exciting new computational algorithm. The rule, “if it’s not broke, don’t fix it” applies to the development of DDSS, as well as other aspects of technology. Developers must carefully define the scope and nature of the process to be automated. They must also understand the process to be automated well enough to reduce the process to an algorithm. All systems, especially DDSS, have boundaries (both in domain coverage and algorithm robustness) beyond which the systems often fail. Developers must understand these limits and make users aware of them. Each algorithm must be studied to determine the ways in which it might fail, due to

both inherent limitations and flaws that might occur during the process of implementation.²

Developers and interested third parties must evaluate any automated system carefully, initially “in vitro” (outside of the patient care arena, with no risks to patients), and, once warranted, in vivo (prospectively, on the front lines of actual patient care delivery) in order to determine if the automated system improves or promotes important outcomes that are not possible with the pre-existing manual system.⁶⁴ Finally, developers and users must demonstrate the practical utility of the system by showing that clinicians can adopt it for productive daily use.² A potentially great system that is not used cannot have a beneficial impact on clinical outcomes. Unfortunately, few, if any, of the existing clinical decision support systems have yet fulfilled these criteria.

There are a number of problems that have limited the ultimate success of DDSS to date. These include: difficulties with domain selection and knowledge-base construction and maintenance; problems with the diagnostic algorithms and user interfaces; the problem of system evolution, including evaluation, testing, and quality control; issues related to machine interfaces and clinical vocabularies; and legal and ethical issues. These issues are discussed below.

Clinical Domain Selection

DDSS domain selection is often problematic. Substantial clinical domains must be chosen in order to avoid creating “toy” systems. However, construction of knowledge bases, to support substantial DDSS, can require dozens of person-years of effort in broad domains such as general internal medicine. To date, although most large medical knowledge bases have at least initially been created in the academic environment, many projects do not have adequate funding to sustain such activity over time.⁶⁵ Availability of adequate domain expertise is also a problem. Clinical collaborators generally earn their wages through patient care or research, and sustaining high-level input from individuals with adequate clinical expertise can be difficult in the face of real-world demands. Commercial vendors must hire an adequate and well qualified staff of physicians in order to maintain medical knowledge bases. However, the income generated through the sale of DDSS programs is limited by the number of users who purchase a program or its updates, so scaling up a DDSS maintenance department can be difficult.

Different problems affect DDSS with narrow domains. One problem is garnering an adequate audience. The CASNET system was an exemplary prototypic system for reasoning pathophysiologically about the diagnosis and therapy of glaucoma.⁶⁶ It typifies a problem that can occur with successful experimental expert systems—the persons most likely to require a specialized system’s use in clinical medicine are the domain experts whose

knowledge was used to develop the system. The persons who routinely diagnose and treat glaucoma are ophthalmologists, who are by definition board-certified specialists in the domain of ophthalmology. The program, in effect, preaches to the choir. It is more difficult for an automated system to provide benefit to experts in that specialty, so experts may not use it, but generalists are also unlikely to use a system with very narrow functioning. A program like the CASNET system must be extremely robust and provide more than one kind of service (e.g., it should have integrated record-management and other functions) in order for it to find use in clinical practice.

Knowledge-Base Construction and Maintenance

Knowledge base maintenance is critical to the clinical validity of a DDSS.¹ Yet it is hard to judge when new clinical knowledge becomes an established “fact.” The first reports of new clinical discoveries in highly regarded medical journals must await confirmation by other groups over time before their content can be added to a medical knowledge base. The nosological labels used in diagnosis reflect the current level of scientific understanding of pathophysiology and disease, and they may change over time without the patient or the patient’s illness, per se, changing.¹ For example, changes occur in how a label is applied when the “gold standard” for making a diagnosis shifts from a pathological biopsy result to an abnormal serological test—patients with earlier, previously unrecognized forms of the illness may be labeled as having the disease. Corresponding changes must be made to keep a DDSS knowledge base up to date.

Knowledge-base construction must be a scientifically reproducible process that can be accomplished by qualified individuals at any site.⁶⁷ Knowledge-base construction should be clinically grounded, based on “absolute” clinical knowledge whenever possible. Attempts to “tune” the DDSS knowledge base to improve performance on a given case or group of cases should be strongly discouraged unless such tuning has an objective basis, such as information culled from the medical literature. If the process of knowledge-base construction is highly dependent on a single individual, or can only be carried out at a single institution, then the survival of that system over time is in jeopardy. While much of the glamour of computer-based diagnostic systems lies in the computer algorithms and interfaces, the long-term value and viability of a system depends on the quality, accuracy, and timeliness of its knowledge base.¹

Even initially successful DDSS cannot survive unless the medical knowledge bases supporting them are kept current. This can require Herculean efforts. Shortliffe’s MYCIN program⁵⁷ was developed as a research project to demonstrate the applicability of rule-based expert systems to clinical medicine. MYCIN was a brilliant, pioneering effort in this regard. The evaluation of MYCIN in the late 1970s by Yu and colleagues demonstrated that

the program could perform at the expert level on challenging cases.⁶⁸ But MYCIN was never put into routine clinical use, nor was an effort made to update its knowledge base over time. After 1979, lack of maintenance caused its antibiotic therapy knowledge base to become out of date.

Diagnostic Decision Support Systems—Diagnostic Algorithms and User Interfaces

Just as computer-based implementation of many complex algorithms involves making trade-offs between space (memory) and time (CPU cycles), development of real-world diagnostic systems involves a constant balancing of theory (model complexity) and practicality (ability to construct and maintain adequate medical databases or knowledge bases, and ability to create systems which respond to users' needs in an acceptably short time interval).⁵⁹ We may understand, in theory, how to develop systems that take into account gradations of symptoms, the degree of uncertainty in the patient and/or physician-user regarding a finding, the severity of each illness under consideration, the pathophysiological mechanisms of disease, and/or the time course of illnesses. Such complexities may ultimately be required to make actual systems work reliably. However, it is not yet practical to build such complex, broad-based systems for patient care. The effort required to build and maintain superficial knowledge bases is measured in dozens of person-years of effort, and more complex knowledge bases are likely to require an order of magnitude greater effort.¹

Although some people believe that the computer will eventually replace the physician,⁶⁹ that position is not very tenable. A clinician cannot convey his or her complete understanding of an involved patient case to a computer program. One can never assume that a computer program "knows" all that needs to be known about the patient case, no matter how much time and effort is spent on data input into the computer system. As a result, the clinician-user who directly evaluated the patient must be considered to be the definitive source of information about the patient during the entire course of any computer-based consultation.² In addition, the highly skilled health-care practitioner—who understands the patient as a person—possesses the most important intellect to be employed during a consultation. That user should intellectually control the process of computer-based consultation. DDSS must be designed to permit users to apply individual tools to assist with the sequence of steps in the diagnostic process in the sequence that the user prefers at the time, not in an arbitrary sequence selected by the DDSS algorithm.

All DDSS, and especially narrowly focused ones, face the "critical mass" problem. Few clinicians are likely to purchase and install office computer systems solely to run one application. The number of narrow DDSS that could be useful in the setting of a primary care practitioner's office is potentially measured in tens or hundreds. Yet few computer-literate individuals learn how to successfully operate more than a dozen applications. Until

there is a standard, integrated environment and user interface that allows smooth transition among dedicated applications, DDSS are not likely to be used heavily. It is possible that the Internet, with a common user interface across multiple hardware platforms, will evolve to be the integrated environment required. However, current limitations of such interfaces leave this an open question. Systems must provide flexible environments that adapt to the user's needs and problems, rather than providing an interface that is inflexible and which penalizes the user for deviating from the normal order of system operation. It must be easy to move from one program function to another if it is common for the healthcare user to do so mentally on their own. Transitions must be facilitated when frequent patterns of usage emerge.

Diagnostic Decision Support Systems Testing, Evaluation, and Quality Control

System evaluation in biomedical informatics should take place as an ongoing, strategically planned process, not as a single event or small number of episodes.^{61,64} Complex software systems and accepted medical practices both evolve rapidly, so evaluators and readers of evaluations face moving targets. As previously noted, systems are of value only when they help users to solve users' problems. Users, not systems, characterize and solve clinical diagnostic problems. The ultimate unit of evaluation should be whether the user plus the system is better than the unaided user with respect to a specified task or problem (usually one generated by the user).

It is extremely important during system development to conduct informal "formative" type evaluations. As a part of this process, new cases must be analyzed with the DDSS on a regular (e.g., weekly) basis. After each failure of the DDSS to make a "correct" diagnosis, careful analysis of both the system's knowledge base and diagnostic algorithms must be carried out. Both the information in the knowledge base on the "correct" diagnosis, and the information on any diagnoses offered in error, must be reviewed and potentially updated. In addition, periodic rerunning of previous series of test cases should be done on an annual (or similar) basis, to verify that there has not been significant "drift" in either the knowledge base or the diagnostic program that would influence the system's abilities.

Formal evaluations of DDSS should take into account the following four perspectives: (1) appropriate evaluation design; (2) specification of criteria for determining DDSS efficacy in the evaluation; (3) evaluation of the boundaries or limitations of the DDSS; and (4) identification of potential reasons for "lack of system effect."⁶⁴ Each of these issues is discussed below.

Appropriate Evaluation Design

Evaluation plans should be appropriate for the information needs being addressed, the level of system maturity, and users' intended form of DDSS usage (or specific system function evaluated).^{62,64} The same DDSS may

serve as an electronic textbook for one user, a diagnostic checklist generator for another user, a consultant to determine the next useful step in a specific patient's evaluation for a third user, and a tool to critique/reinforce the users' own pre-existing hypotheses for a fourth user. Each system function would require a different form of evaluation whenever anticipated user benefits depend on which system function is used. Evaluations should clearly state which user objective is being studied and which of the available system functions are relevant to that objective.

In 1994, Berner and colleagues evaluated the ability of several systems to generate first-pass differential diagnoses from a fixed set of input findings.⁷⁰ These findings were not generated by everyday clinical users, but from written case summaries of real patient data. That approach was dictated by the desire to standardize system inputs and outputs for purposes of multisystem use. The primary goal of Berner et al. was to develop methods and metrics that would characterize aspects of system performance in a manner useful for rationally comparing different systems and their functions. All of the systems in that study were capable of generating questions to further refine the initial differential diagnoses, which is the intended mode of clinical use for such systems. Because that study was not intended to produce a definitive rating or comparison of the systems themselves, the involved systems were not placed in the hands of end users, nor were the systems used in a manner to address common end-user needs. Even though the evaluation did not examine this capability, the methods used by Berner were sound. Generating a first-pass differential diagnosis is a good initial step, but subsequent evidence gathering, reflection, and refinement are required.

There are important questions that must be answered in the evaluation. Are the problems ones that clinical users generate during clinical practice, or artificial problems generated by the study design team? Is the case material accurately based on actual patient cases? Note that there can be no truly verifiable diagnosis when artificial, manually constructed or computer-generated cases are used. Are the evaluation subjects clinical users whose participation occurs in the clinical context of caring for the patients used as "test cases?" Are clinical users evaluating abstracts of cases they have never seen, or are nonclinical personnel evaluating abstracted clinical cases using computer systems? Are users free to use all system components in whatever manner they choose, or is it likely that the study design will constrain users to exercise only limited components of the system? The answers to these questions will determine the generalizability of the results of the evaluation.

Specification of Criteria for Determining Diagnostic Decision Support Systems Efficacy in the Evaluation

Evaluations must determine if the criteria for "successful" system performance are similar to what clinical practitioners would require during actual practice. Diagnosis itself, or more properly, "diagnostic benefit," must be

defined in such contexts. Similarly, what it means to establish a diagnosis must be carefully defined. For example, it is not adequate to accept hospital discharge diagnoses at face value as a “gold standard” since discharge diagnoses are not of uniform quality—they have been documented to be influenced by physician competency, coding errors, and economic pressures. Furthermore, some discharge diagnoses may be “active” (undiagnosed at admission and related to the patient’s reason for hospitalization), while others may be relevant but inactive. Criteria for the establishment of a “gold standard” diagnosis should be stated prospectively, before beginning data collection.

Evaluation of the Boundaries or Limitations of the Diagnostic Decision Support Systems

A system may fail when presented with cases outside its knowledge-base domain, but if an evaluation uses only cases from within that domain, this failure may never be identified. The limits of a system’s knowledge base are a concern because patients do not accurately triage themselves to present to the most appropriate specialists. For instance, as discussed earlier, de Dombal’s abdominal pain system performed very well when used by surgeons to determine if patients presenting with abdominal pain required surgery. However, a patient with atypical appendicitis may present to an internist, and a patient with abdominal pain due to lead poisoning may first see a surgeon.

Identification of Potential Reasons for “Lack of System Effect”

DDSS operate within a system that not only includes the DDSS itself, but also the user and the healthcare environment in which the user practices. A model of all of the possible influences on the evaluation outcomes would include DDSS-related factors (knowledge-base inadequacies, inadequate synonyms within vocabularies, faulty algorithms, etc.), user-related factors (lack of training or experience with the system, failure to use or understand certain system functions, lack of medical knowledge or clinical expertise, etc.) and external variables (lack of available gold standards, failure of patients or clinicians to follow-up during study period). It is important to recognize that studies that focus on one aspect of system function may have to make compromises with respect to other system or user-related factors in order to have an interpretable result. Additionally, in any DDSS evaluation, the user’s ability to generate meaningful input into the system, and the system’s ability to respond to variable quality of input from different users, is an important concern.

Evaluations of DDSS must each take a standard objective (which may be only one component of system function) and measure how effectively the system enhances users’ performances, using a study design that incorporates the most appropriate and rigorous methodology relative to the stage of system development. The ultimate clinical end user of a given DDSS must determine if published evaluation studies examine the system’s

function in the manner that the user intends to use it. This is analogous to a practitioner determining if a given clinical trial (of an intervention) is relevant to a specific patient by matching the given patient's characteristics to the study's inclusion and exclusion criteria, population demographics, and the patient's tolerance for the proposed forms of therapy as compared to alternatives. The reporting of an individual "negative study" of system performance should not, as it often does now, carry the implication that the system is globally suboptimal. A negative result for one system function does not mean that, for the same system, some users cannot derive significant benefits for other system functions. Similarly, complete evaluation of a system over time should examine basic components (e.g., the knowledge base, ability to generate reasonable differential diagnoses, ability to critique diagnoses, and so on), as well as clinical functionality (e.g., can novice users, after standard training, successfully employ the system to solve problems that they might not otherwise solve as efficiently or completely?). The field of DDSS evaluation will become mature only when clinical system users regularly derive the same benefit from published DDSS evaluations as they do from evaluations of more standard clinical interventions.

Diagnostic Decision Support Systems Interface and Vocabulary Issues

A critical issue for the success of large-scale, generic DDSS is their environment. Small, limited, "niche" systems may be adopted and used by the focused community for which they are intended, while physicians in general medical practice, for whom the large-scale systems are intended, may not perceive the need for diagnostic assistance on a frequent enough basis to justify purchase of one or more such systems. Therefore, it is common wisdom that DDSS are most likely to succeed if they can be integrated into a clinical environment so that patient data capture is already performed by automated laboratory and/or hospital information systems. In such an environment, the physician will not have to manually enter all of a patient's data in order to obtain a diagnostic consultation. However, it is not straightforward to transfer the information on a patient from a hospital information system to a diagnostic consultation system. If 100 hematocrits were measured during a patient's admission, which one(s) should be transferred to the consultation system—the mean, the extremes, or the value typical for a given time in a patient's illness? Should all findings be transferred to the consultation system, or only those findings relevant to the patient's current illness? These questions must be resolved by careful study before one can expect to obtain patient consultations routinely and automatically within the context of a hospital information system. Another reason for providing an integrated environment is that users will not use a system unless it is sufficiently convenient to do so. By integrating DDSS into healthcare provider results reporting and order entry systems, the usual computer-free workflow processes of the clinician can be replaced

with an environment conducive to accomplishing a number of computer-assisted clinical tasks, making it more likely that a DDSS will be used.

Interfaces between automated systems are, at times, as important as the man-machine interface. Fundamental questions, such as the definition of diseases and of findings, limit our ability to combine data from the literature, from clinical databanks, from hospital information systems, and from individual experts' experiences in order to create DDSS. Similar problems exist when trying to match the records from a given case (collected manually or taken from an electronic medical record) with a computer-based diagnostic system. A diagnostic system may embody different definitions for patient descriptors than those of the physician who evaluated the patient, even though the words used by each may be identical.

In order to facilitate data exchange among local and remote programs, it is mandatory to have a lexicon or interlingua which facilitates accurate and reliable transfer of information among systems that have different internal vocabularies (data dictionaries). The United States National Library of Medicine Unified Medical Language System (UMLS) project, which started in 1987 and continues through the present time, represents one such effort.⁷¹

Legal and Ethical Issues

Proposals have been made for governmental agencies, such as the United States Food and Drug Administration (FDA), which oversees medical devices, to regulate use of clinical software programs such as DDSS. These proposals include a variety of recommendations that manufacturers of such systems would be required to perform to guarantee that the systems would function per specifications.

There is debate about whether these consultation systems are actually devices in the same sense as other regulatable devices. In the past, governmental regulation has not been considered necessary when a licensed practitioner is the user of a DDSS.⁷² It would be both costly and difficult for the government to regulate DDSS more directly, even if a decision were made to do so. For general DDSS programs like Iliad, QMR, Meditel and DXplain, with hundreds to thousands of possible diagnoses represented in their knowledge bases,⁷⁰ conducting prospective clinical trials, to demonstrate that the system worked for all ranges of diagnostic difficulty for a variety of patients with each diagnosis, would require enrollment of huge numbers of patients and would cost millions of dollars.

Other approaches, such as a "software quality audit" to determine, prospectively, if a given software product has flaws, would also be clinically impractical. The clinician seeking help may have any of several dozen kinds of diagnostic problems in any given case. Unless it is known, for a given case, which kind of problem the practitioner will have, performing a software quality audit could not predict if the system would be useful.

Consider the dilemma the FDA or other responsible regulatory agency would face if it agreed to review situations when a user files a complaint. First, one must note that few patients undergo definitive enough diagnostic evaluations to make it possible to have a “gold standard” (certain) diagnosis. So if the doctor claims the program was wrong, a major question would be how governmental auditors would know what the actual “right” diagnosis was. Second, the reviewers would need to know all of the information that was knowable about the patient at the time the disputed diagnosis was offered. This could potentially violate patient confidentiality if the records were sent to outsiders for review. All sources of information about the patient would have to be audited, and this could become as difficult as evidence gathering in a malpractice trial. To complete the sort of audit described, the governmental agency would have to determine if the user had been appropriately trained and if the user used the program correctly. Unless the program had an internally stored complete audit trail of each session (down to the level of saving each keystroke the user typed), the auditors might never be able to recreate the session in question. Also, the auditors would have to study whether the program’s knowledge base was appropriate. Initial development of the R-QMR knowledge base at the University of Pittsburgh required an average of three person-weeks of a clinician’s time, which went into literature review of 50–150 primary articles about each disease, with additional time for synthesis and testing against cases of real patients with the disease. For an auditor to hire the required expertise to review this process for hundreds to thousands of diseases for each of the programs that it would have to review and subsequently monitor would be costly and cumbersome. The ultimate question, very difficult to answer, would be whether the original user in the case in question used the system in the best way possible for the given case. Making such a determination would require the governmental agency to become expert in the use of each DDSS program. This could take up to several months of training and practice for a single auditor to become facile in the use of a single system. It would be difficult for a governmental agency to muster the necessary resources for even a small number of such complaints, let alone nationwide for multiple products with thousands of users. The complexity of these issues makes it very difficult to formulate appropriate regulatory policy. In addition to legal issues concerning regulation, there are other legal and ethical issues relating to use of DDSS that are discussed in Chapter 6.

The Future of Diagnostic Decision Support Systems

It is relatively safe to predict that specialized, focused DDSS will proliferate, and a sizable number of them will find widespread application.¹ As new medical devices are developed and older devices automated, DDSS software that enhances the performance of the device, or helps users to interpret the output of the device, will become essential. Computerized

electrocardiogram (EKG) analysis, automated arterial blood gas interpretation, automated protein electrophoresis reports, and automated differential blood cell counters, are but a few examples of such success at the present time. In fact, since Miller published an article summarizing past DDSS activities in 1994,¹ the great majority of the several dozen articles on “diagnosis, computer-assisted” indexed in MEDLINE have described focused systems for the interpretation of images (radiological studies and pathology cytology/sections/slides), signals (EKGs, electroencephalograms (EEGs), and so on), and diagnosis of very narrowly defined clinical conditions. One by-product of the success of these systems is that users may be less vigilant in questioning system accuracy. In a recent article, Tsai and colleagues pointed out the potential clinical dangers of overreliance of inexperienced clinicians on computer systems for advice—they tend to follow the advice even when it is wrong.⁷³

The future of large-scale, “generic” diagnostic systems is hopeful, although less certain. As discussed in this and other chapters, a number of major challenges remain to be solved before DDSS that address large medical problem domains can succeed over time. No matter what the level of use of large-scale, generic DDSS in clinical practice, it is well established that such systems can play a valuable role in medical education.¹ The process of knowledge-base construction, utilization of such knowledge bases for medical education in the form of patient case simulations, and the use of DDSS have all been shown to be of educational value in a variety of institutional settings.

One exception to the overwhelming recent trend of developing focal DDSS has been the development of the ISABEL system for general diagnosis.^{74–78} Berner discussed the implications of evaluating DDSS using less than absolute gold standards, as was proposed by the ISABEL team, in a well balanced perspective covering “correctness” of diagnosis, “appropriateness” of management suggestions, end-user acceptance and satisfaction, degree of adoption and use of a DDSS, and issues related to human-computer system interfaces.⁷⁹

In summary, the future of DDSS appears to be promising. The number of researchers in the field is growing. The diversity of DDSS is increasing. The number of commercial enterprises interested in DDSS is expanding. Rapid improvements in computer technology continue to be made. A growing demand for cost-effective clinical information management, and the desire for better health care, is sweeping the United States.⁸⁰ Evidence-based medicine is now in vogue. All these factors will insure that new and productive DDSS applications will be developed, evaluated, and used.

References

1. Miller RA. Medical diagnostic decision support systems—past, present, and future: a threaded bibliography and commentary. *J Am Med Inform Assoc* 1994;1:8–27.

2. Miller RA. Why the standard view is standard: people, not machines, understand patients' problems. *J Med Philos* 1990;15:581–591.
3. Flexner SB, Stein J, eds. *The Random House college dictionary*, revised edition. New York: Random House, Inc.; 1988:366.
4. Osheroff JA, Forsythe DE, Buchanan BG, Bankowitz RA, Blumenfeld BH, Miller RA. Physicians' information needs: an analysis of questions posed during clinical teaching in internal medicine. *Ann Intern Med* 1991;114:576–581.
5. Forsythe DE, Buchanan BG, Osheroff JA, Miller RA. Expanding the concept of medical information: an observational study of physicians' information needs. *Comput Biomed Res* 1992;25:181–200.
6. Miller R, Masarie FE, Myers J. Quick Medical Reference (QMR) for diagnostic assistance. *MD Comput* 1986;3:34–38.
7. Miller RA, Masarie FE Jr. The quick medical reference (QMR) relationships function: description and evaluation of a simple, efficient "multiple diagnoses" algorithm. *Medinfo* 1992:512–518.
8. Miller RA, Masarie FE Jr. The demise of the "Greek Oracle" model for medical diagnosis systems. *Methods Inf Med* 1990;29:1–2.
9. de Dombal FT, Leaper DJ, Horrocks JC, Staniland JR, McCann AP. Human and computer-aided diagnosis of abdominal pain: further report with emphasis on performance of clinicians. *Br Med J* 1974;1:376–380.
10. Adams ID, Chan M, Clifford PC, et al. Computer aided diagnosis of acute abdominal pain: a multicentre study. *Br Med J (Clin Res Ed)* 1986;293:800–804.
11. Rosenbaum M, Leibel RL, Hirsch J. Obesity. *N Engl J Med* 1997;337:396–407.
12. Newell A, Simon HA. *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall; 1972.
13. Kahneman D, Slovic P, Tversky A, eds. *Judgment under uncertainty: heuristics and biases*. Cambridge, UK: Cambridge University Press; 1982.
14. Elstein AS, Shulman LS, Sprafka SA. *Medical problem solving: an analysis of clinical reasoning*. Cambridge, MA: Harvard University Press; 1978.
15. Kassirer JP, Gorry GA. Clinical problem-solving—a behavioral analysis. *Ann Intern Med* 1978;89:245–255.
16. Miller GA. The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychol Rev* 1956;63:81–97.
17. Evans DA, Patel VL, eds. *Cognitive science in medicine*. Cambridge, MA: MIT Press; 1989.
18. Pople HE Jr. Heuristic methods for imposing structure on ill-structured problems: the structuring of medical diagnostics. In: Szolovits P, ed. *Artificial intelligence in medicine*. AAAS Symposium Series. Boulder, CO: Westview Press. 1982;119–190.
19. Simon HA. The structure of ill-structured problems. *Artif Intell* 1973;4:181–201.
20. Miller RA, Pople HE Jr, Myers J. Internist-I, an experimental computer-based diagnostic consultant for general internal medicine. *N Engl J Med* 1982;307:468–476.
21. Eddy DM, Clanton CH. The art of diagnosis: solving the clinicopathological conference. *N Engl J Med* 1982;306:1263–1269.
22. Weed LL. Medical records that guide and teach. *N Engl J Med* 1968;278:593–600; 652–657.

23. Hurst JW, Walker HK, eds. *The problem-oriented system*. New York: Medcom Learning Systems; 1972.
24. Kassirer JP. Diagnostic reasoning. *Ann Intern Med* 1989;110:893–900.
25. Brunswik E. Representative design and probabilistic theory in a functional psychology. *Psychol Rev* 1955;62:193–217.
26. Ledley RS, Lusted LB. Reasoning foundations of medical diagnosis; symbolic logic, probability, and value theory aid our understanding of how physicians reason. *Science*. 1959;130:9–21.
27. Warner HR, Toronto AF, Veasey LG, Stephenson R. Mathematical approach to medical diagnosis. *JAMA* 1961;177:75–81.
28. Raiffa H. *Decision analysis*. Reading, MA: Addison-Wesley Inc.; 1970.
29. Pauker SG, Kassirer JP. *Decision analysis*. *N Engl J Med* 1987;316:250–258.
30. Dawes RM, Faust D, Meehl PE. Clinical versus actuarial judgment. *Science* 1989;243:1668–1674.
31. Gigerenzer G, Hoffrage U. How to improve Bayesian reasoning without instruction: frequency formats. *Psychol Rev* 1995;102:684–704.
32. Feinstein AR. An analysis of diagnostic reasoning. I. The domains and disorders of clinical macrobiology. *Yale J Biol Med* 1973;46:212–232.
33. Feinstein AR. An analysis of diagnostic reasoning. II. The strategy of intermediate decisions. *Yale J Biol Med* 1973;46:264–283.
34. Horvitz EJ, Heckerman DE. The inconsistent use of measures of certainty in artificial intelligence research. In: *Uncertainty in Artificial Intelligence*, Vol. 1. LN Kanal and JF Lemmer, eds. Amsterdam; New York: Elsevier Science; 1986: 137–151.
35. Blois MS. Clinical judgment and computers. *N Engl J Med* 1980;303:192–197.
36. Shortliffe EH, Buchanan BG, Feigenbaum EA. Knowledge engineering for medical decision-making: a review of computer-based clinical decision aids. *Proc IEEE* 1979;67:1207–1224.
37. Willems JL, Abreu-Lima C, Arnaud P, et al. The diagnostic performance of computer programs for the interpretation of electrocardiograms. *N Engl J Med* 1991;325:1767–1773.
38. Bleich HL. Computer evaluation of acid-base disorders. *J Clin Invest* 1969;48:1689–1696.
39. Gorry GA, Barnett GO. Experience with a model of sequential diagnosis. *Comput Biomed Res* 1968;1:490–507.
40. Szolovits P, Pauker SG. Categorical and probabilistic reasoning in medical diagnosis. *Artif Intell* 1978;11:114–144.
41. Lipkin M, Hardy JD. Differential diagnosis of hematological diseases aided by mechanical correlation of data. *Science* 1957;125:551–552.
42. Lipkin M, Hardy JD. Mechanical correlation of data in differential diagnosis of hematological diseases. *JAMA* 1958;166:113–123.
43. Lipkin M, Engle RL Jr, Davis BJ, Zworykin VK, Ebald R, Sendrow M. Digital computer as aid to differential diagnosis. *Arch Intern Med* 1961;108:56–72.
44. Lindberg DAB, Rowland LR, Buch CR Jr, Morse WF, Morse SS. Consider: A computer program for medical instruction. In: *Proceedings of the Ninth IBM Medical Symposium*. White Plains, NY: IBM, 1968. Conference dates: 9th IBM Medical Symposium. Burlington, Vermont, October 24–26, 1969.
45. Nelson SJ, Blois MS, Tuttle MS, et al. Evaluating RECONSIDER: a computer program for diagnostic prompting. *J Med Syst* 1985;9:379–388.

46. Weiss S, Kulikowski CA. EXPERT: a system for developing consultation models. Proceedings of the Sixth International Joint Conference on Artificial Intelligence; Tokyo, Japan, 1979.
47. Lindberg DAB, Sharp GC, Kingsland LC, III et al. Computer-based rheumatology consultant. In: Linberg DAB, Kaihara S, eds. Proceedings of MEDINFO 80 Tokyo, Third World Conference on Medical Informatics, Amsterdam: North Holland Publishing Company, 1980:1311–1315.
48. Moens HJ, van der Korst JK. Development and validation of a computer program using Bayes' theorem to support diagnosis of rheumatic disorders. *Ann Rheum Dis* 1992;51:266–271.
49. Gorry A. Strategies for computer-aided diagnosis. *Math Biosci* 1968;2:293–318.
50. Pauker SG, Gorry GA, Kassirer JP, Schwartz WB. Towards the simulation of clinical cognition. Taking a present illness by computer. *Am J Med* 1976;60:981–996.
51. Waxman HS, Worley WE. Computer-assisted adult medical diagnosis: subject review and evaluation of a new microcomputer-based system. *Medicine* 1990;69:125–136.
52. Pople HE, Myers JD, Miller RA. DIALOG: A model of diagnostic logic for internal medicine. Proceedings of the Fourth International Joint Conference on Artificial Intelligence; Tbilisi, Georgia, 1975:848–855.
53. First MB, Soffer LJ, Miller RA. QUICK (Quick Index to Caduceus Knowledge): using the Internist-I/Caduceus knowledge base as an electronic textbook of medicine. *Comput Biomed Res* 1985;18:137–165.
54. Miller RA, McNeil MA, Challinor S, Masarie FE Jr, Myers JD. Status report: the Internist-1/Quick Medical Reference project. *West J Med* 1986;145:816–822.
55. Hupp JA, Cimino JJ, Hoffer EF, Lowe HJ, Barnett GO. DXplain—a computer-based diagnostic knowledge base. In: Proceedings of the Fifth World Conference on Medical Informatics (MEDINFO 86). Amsterdam: North-Holland, pp. 117–121.
56. Warner HR, Haug P, Bouhaddou O, Lincoln, M. ILIAD as an expert consultant to teach differential diagnosis. In RA Greenes ed. Proceedings of the Twelfth Annual Symposium on Computer Applications in Medical Care. Los Angeles, CA: IEEE Computer Society, 1988:371–376.
57. Shortliffe EH. Computer-based medical consultations: MYCIN. New York: Elsevier; 1976.
58. Miller PL. A critiquing approach to expert computer advice: ATTENDING. Boston: Pittman; 1984.
59. Aliferis CF, Miller RA. On the heuristic nature of medical decision support systems. *Methods Inf Med* 1995;34:5–14.
60. Reggia JA, Nau DS, Wang PY. Diagnostic expert systems based on a set covering model. *Int J Man Mach Stud* 1983;19:437–460.
61. Berman L, Miller RA. Problem area formation as an element of computer aided diagnosis: a comparison of two strategies within quick medical reference (QMR). *Methods Inf Med* 1991;30:90–95.
62. Stead WW, Haynes RB, Fuller S, et al. Designing medical informatics research and library-resource projects to increase what is learned. *J Am Med Inform Assoc* 1994;1:28–33.
63. Covell DG, Uman GC, Manning PR. Information needs in office practice: are they being met? *Ann Intern Med* 1985;103:596–599.

64. Miller RA. Evaluating evaluations of medical diagnostic systems. *J Am Med Informatics Assoc* 1996;3:429–431.
65. Yu VL. Conceptual obstacles in computerized medical diagnosis. *J Med Philos* 1983;8:67–75.
66. Weiss S, Kulikowski CA, Safir A. Glaucoma consultation by computer. *Comput Biol Med* 1978;8:24–40.
67. Giuse NB, Giuse DA, Miller RA, et al. Evaluating consensus among physicians in medical knowledge base construction. *Methods Inf Med* 1993;32:137–145.
68. Yu VL, Fagan LM, Wraith SM, et al. Antimicrobial selection by computer: a blinded evaluation by infectious disease experts. *JAMA* 1979;242:1279–1282.
69. Mazoue JG. Diagnosis without doctors. *J Med Philos* 1990;15:559–579.
70. Berner ES, Webster GD, Shugerman AA, et al. Performance of four computer-based diagnostic systems. *N Engl J Med* 1994;330:1792–1796.
71. Lindberg DA, Humphreys BL, McCray AT. The unified medical language system. *Methods Inf Med* 1993;32:281–291.
72. Young FE. Validation of medical software: present policy of the Food and Drug Administration. *Ann Intern Med* 1987;106:628–629.
73. Tsai TL, Fridsma DB, Gatti G. Computer decision support as a source of interpretation error: the case of electrocardiograms. *J Am Med Inform Assoc*. 2003;10:478–483.
74. Greenough A. Help from ISABEL for paediatric diagnoses. *Lancet*. 2002;360:1259.
75. McKenna C. New online diagnostic tool launched to help doctors. *BMJ* 2002;324:1478.
76. Thomas NJ. ISABEL. *Crit Care*. 2003;7:99–100.
77. Ramnarayan P, Tomlinson A, Rao A, Coren M, Winrow A, Britto J. ISABEL: a web-based differential diagnostic aid for paediatrics: results from an initial performance evaluation. *Arch Dis Child*. 2003;88:408–413.
78. Ramnarayan P, Kapoor RR, Coren M, et al. Measuring the impact of diagnostic decision support on the quality of decision-making: development of a reliable and valid composite score. *J Am Med Inform Assoc*. 2003;10:563–572.
79. Berner E. Diagnostic decision support systems: how to determine the gold standard? *J Am Med Inform Assoc*. 2003;10:608–610.
80. Osheroff JA, Teich JM, Middleton BF, Steen EB, Wright A, Detmer DE. A roadmap for national action on clinical decision support. For AMIA as a work product for ONC contract HHSP23300500877P. June 13, 2006. Available from <http://www.amia.org/inside/initiatives/cds/cdsroadmap.pdf>. Accessed on July 8, 2006.

6

Ethical and Legal Issues in Decision Support

KENNETH W. GOODMAN

Discrete maladies or illnesses tend to produce particular signs and symptoms. This natural correlation makes possible the process of diagnosis and prognosis. In fact, so strong is our belief in the regularity of signs and symptoms that the process has long been regarded as straightforward, if not easy: “. . . there is nothing remarkable,” Hippocrates suggested some 2,400 years ago, “in being right in the great majority of cases in the same district, provided the physician knows the signs and can draw the correct conclusions from them”.¹

Of course, accurate diagnosis and prognosis can be quite difficult, even given the regularity of signs and symptoms. For one thing, “knowing the signs” requires a great deal of empirical knowledge and experience. For another, there is rarely a unique and isomorphic relationship between symptom and disease. Significantly, Hippocrates smuggles into his account a presumption of the very thing being described. To say that being right is unremarkable when one can draw the “correct conclusions” is to say that it is easy to be right when you know how to be right. Or, making an accurate diagnosis or prognosis is easy if one knows how to make an accurate diagnosis or prognosis!

The need to make accurate diagnoses is not based merely on the personal satisfaction that comes from being right, as gratifying as that is. It is based on the good effects that follow more frequently from accurate diagnoses than from inaccurate diagnoses. It is also based on the bad effects that error entails.

In the context of trust and vulnerability that shape patient-physician and patient-nurse encounters, there emerges an ethical imperative: to adhere to, or surpass, educational and professional standards, to monitor changes in one's domain, to know when one is out of one's depth. Decision support systems have the potential to assist clinicians, but their use also entails a number of ethical concerns. In fact, this is evidence for the maturity of the science: new health technologies almost always raise ethical issues, and it should come as no surprise that clinical decision support would provide a number of challenges for those who use, or would use, computers to assist,

guide or test clinical decisions. Any comprehensive treatment of clinical decision support systems (CDSS) should include a review of ethical issues. In what follows, we identify a number of ethical issues and positions that emerge when intelligent machines are used to perform or support clinical decision making, and we survey key legal and regulatory issues.

Ethical Issues

Background and Current Research

It has been clear for more than a decade that health computing raises interesting and important ethical issues. In a crucial early contribution, a physician, a philosopher, and a lawyer identified a series of ethical concerns, not the least of which are several surrounding the questions of who should use a “medical computer program” and under what circumstances.² Another early contribution emphasized the challenges raised by threats to physician autonomy.³

What has emerged since has been called the “Standard View” of computational diagnosis.⁴ Randolph A. Miller, M.D., a key figure both in the scientific evolution of computational decision support and in scholarship on correlate ethical issues, has argued that “Limitations in man-machine interfaces, and, more importantly, in automated systems’ ability to represent the broad variety of concepts relevant to clinical medicine, will prevent ‘human-assisted computer diagnosis’ from being feasible for decades, if it is at all possible.”⁴ Another way of putting this is to say that computers cannot, either in principle or at least for the foreseeable future, supplant human decision makers. This observation entails ethical obligations, namely that computers ought not to be relied on to do what humans do best, and that a “computer diagnosis” cannot, as a matter of course or policy, be allowed to trump a human decision or diagnosis.

Happily, the Standard View has been advanced not by those hostile to the development and use of CDSS, but by leading proponents. The Standard View bespeaks a conservative and cautious approach to applications of a new technology, and, as such, captures important moral intuitions about technological change, risks, and standards.

Interest in the three-way intersection of ethics, medicine, and computing has increased significantly since initial efforts to explore these issues. On the one hand, professional societies such as the American Association for the Advancement of Science, the American College of Physicians and the American Medical Informatics Association have encouraged educational programs and other professional activities. On the other hand, the literature exploring this intersection has progressed significantly, and now includes the first book devoted to the topic.⁵

Three core areas of ethical concern have emerged in discussions of computer systems that are used to remind, consult, or advise clinicians:

(1) care standards; (2) appropriate use and users; and (3) professional relationships.⁶

Care Standards

We know a great deal about responsibility in medicine and nursing. For instance, we know that practitioners should generally not deceive their patients. We know that patients can be especially vulnerable, and that such vulnerability should be respected. And we know that physicians and nurses have a responsibility to do their best, irrespective of economic (dis)incentives, and that they should not attempt treatments that are beyond their training or expertise.

Learning how to meet these and other responsibilities in the context of a broad variety of social problems is arguably the leading task in bioethics. We must first ask whether computing tools help or hinder attempts to meet responsibilities, and, second, whether the tools impose new or special responsibilities. The overarching question may be put thus: does the new technology improve patient care? If the answer is affirmative, we may suppose we have met an important responsibility. If the answer is negative, it seems clear we should not use the new technology. The problem is, we often do not know how to answer the question. That is, we are sometimes unsure whether care will be improved by the use of new technologies. If we want to meet the responsibility to avoid harm, for instance, we are impotent until we can determine the effects of the technology (see Chapter 7). The upshot here is that error avoidance is an ethical imperative, both to maximize positive, short-term consequences and to ensure that, in the long run, informatics is not associated with error or carelessness, or the kind of cavalier stance sometimes associated with high-tech boosterism.

The concept of error avoidance is wed to that of a standard of care. Standards evolve in the health professions because they plot the kinds of actions that are most successful in achieving certain ends. To fail to adhere to a standard is thus to increase the risk of error, at least in a mature science. Because errors or their consequences are generally regarded as harms or evils, the obligation to hew to standards is an ethical one.

But standards are empirical constructs, and so are open to revision. New evidence forces changes in standards. (This demonstrates why clinicians have an ethical obligation to monitor the scientific maturation of their disciplines by reading journals, attending conferences, etc.) To be sure, the precise content of any standard might be open to dispute. The “reasonable person” standard requires the postulation of a vague entity; this is particularly problematic when reasonable people disagree, as is often the case in medicine and nursing. A “community standard” similarly fails to identify a bright line between error and success in all circumstances in which it might be invoked. Note also that it is not always bad to forgo adherence to a

practice standard—the standard will generally be invoked in ethical and legal contexts only when there is a bad outcome, or a flagrant disregard for the risk of a bad outcome. Sometimes there are good reasons to violate a standard. This demonstrates how some clinical progress is possible: if everyone in all cases stuck to a rigid standard, there would be no internal evidence to support modifications of the standard. In other cases, standards are modified as a result of clinical trial findings, observational studies, and serendipitous discoveries.

In the case of computer-assisted diagnoses, the challenge is perhaps best put in the form of a question: does use of a decision support system increase the risk of error? Note in this regard the following three points. First, while accurate diagnosis is often linked to optimal treatment, this is not always the case: some patients are treated appropriately despite an inaccurate diagnosis, and some are treated incorrectly despite an accurate diagnosis. Second, one might still be able to provide an optimal treatment with a vague or imprecise diagnosis.⁷ Third, computers can render diagnoses (or perform diagnosis-like functions) outside of clinical contexts, as, for instance, in tests for blood-borne pathogens,⁸ cytology screens,⁹ and the like.

To ask if a computer diagnosis increases (or decreases) the risk of diagnostic or other error is in part to ask whether it will improve patient care. If the answer is that, on balance, the tool increases (the risk of) diagnostic error, then we should say it would be unethical to use it. Significantly, though, what is sought here is an empirical finding or a reasoned judgment—where such a finding is often lacking or even methodologically hard to come by, or where such a judgment is based on inadequate epistemic support, at least according to standards otherwise demanded to justify clinical decisions.

This means that we are pressed to answer an ethical question (is it acceptable to use a decision support system?) in a context of scientific uncertainty (how accurate is the system?). Many challenges in contemporary bioethics share this feature, namely, that moral uncertainty parallels scientific or clinical ignorance.

What we generally want in such cases is a way to stimulate the appropriate use of new technologies without increasing patient risk. One approach to doing this is given the nearly oxymoronic term “progressive caution.” The idea is this: “Medical informatics is, happily, here to stay, but users and society have extensive responsibilities to ensure that we use our tools appropriately. This might cause us to move more deliberately or slowly than some would like. Ethically speaking, that is just too bad.”¹⁰ Such a stance attempts the ethical optimization of decision-support use and development by encouraging expansion of the field, but with appropriate levels of scrutiny, oversight, and, indeed, caution.

The moral imperative of error avoidance is, in other words, not antiprogessive. Rather, it is part of a large and public network of checks and balances that seeks to optimize good outcomes by regulating conflicts between

boosters and naysayers. The idea of progressive caution is an attempt to capture the core values of that regulation.

It has been clear since the first efforts to address ethical issues in medical informatics that as computers help the sciences of medicine and nursing to progress, they will also contribute to changes in the standard of patient care. When that happens, however, it increases the likelihood that computer use will come to be required of clinicians. Put differently, in a comparatively short time, there has been a major shift in the availability and use of informatics tools. To the degree that informatics can improve the practice of the health professions, there is a requirement that its tools be used.

This point is often the most disturbing for practitioners. It is troublesome that one might have an obligation to use a tool that has been presented as controversial and in need of further validation. But there is no contradiction here. In fact, it appears that the rise of medical informatics parallels the emergence of other exciting and controversial tools, ranging from organ transplantation techniques and advanced life support to laparoscopic surgical procedures and genetic testing and therapy. It is often the case in history that progress involves this tension. What is wanted is evidence that people of good will can both advance science and safeguard against abuses. Research studies that examine not just the accuracy of the systems, but how they are used, are crucial to collecting that evidence.

Appropriate Use and Users

One way to abuse a tool is to use it for purposes for which it is not intended. Another is to use a tool without adequate training. A third way is to use a tool incorrectly (carelessly, sloppily, etc.) independently of other shortcomings.

There are a number of reasons why one should not use tools in unintended contexts. First, a tool designed for one purpose has a greater likelihood of not working, or not working well, for other purposes. To be sure, one might successfully perform an appendectomy with a kitchen knife, or dice vegetables with a scalpel, but it is bizarre to suggest that one should try either, except in an emergency. A medical computer system may be used inappropriately if, for instance, it was designed for educational purposes but relied on for clinical decision support; or developed for modest decision support (identifying a number of differential diagnoses) but used in such a way as to cause a practitioner to abandon a diagnosis arrived at by sound clinical methods.

In ethically optimizing the use of CDSS, it is perhaps reassuring to know that we have many models and precedents. From advanced life support and organ transplantation to developments in pharmacotherapy and genetics, society regularly has had to cope with technological change in the health

sciences. Managing change requires that new tools are used appropriately and by adequately qualified practitioners. Education is at the core of such management.

Identifying qualifications and providing training must be key components of any movement to expand the use of decision support software. Ethical concerns arise when we are unsure of the appropriate or adequate qualifications and levels of training.⁶

The fear is that: (1) a healthcare novice, or (2) a healthcare professional ignorant of a system's design or capacity will use a decision support system in patient care. The reason the former is worthy of concern is that, as above, the practice of medicine and nursing remain human activities. A nonphysician or nonnurse cannot practice medicine or nursing, no matter how much computational support is available. This is also a concern in the context of consumer health informatics, or the widespread availability of online health advice to the untrained (see Chapter 11). What this means is that the novice might not know when the system is in error or producing flawed output, when it is operating on insufficient information, when it is being used in a domain for which it was not designed, and so on.

There are several reasons we must also focus ethical attention on the use of decision support software by computationally naive health professionals. Such professionals might not use such software to good effect (either by over- or underestimating its abilities), might not be using it properly, or, like the novice, might not know when the system is being used in inappropriate contexts.

Such fears can be addressed by requirements that users of CDSS have appropriate qualifications and be adequately trained in the use of the systems. Unfortunately, it is not yet clear what those qualifications should be or how extensive a training program would be adequate. It is clear, however, that the use of diagnostic software cannot, in the long run, advance ethically without a better sense of where to establish guideposts for qualifications and training. This will be an increasingly important area of research in coming years.

A further ethical concern about appropriate use and users emerges from the potential to deploy decision support systems in contexts of practice evaluation, quality assessment, reimbursement for professional services, and the like. One can imagine an insurance company or managed care organization using decision support to evaluate, or even challenge, clinical decisions. What makes such use problematic is precisely the same ensemble of concerns that led us to disdain applications in other contexts: the primacy of human cognitive expertise, uncertainty about adequate qualifications, and doubt about the consequences for improved patient care. This is not to say that a machine cannot give a correct answer in a particular case but, rather, that there are inadequate grounds to prefer machine decisions as a matter of general policy.

Professional Relationships

Many patients believe, mistakenly, that their physicians are omniscient. Many physicians believe, mistakenly, that their patients are ignoramuses. Recognition of these mistakes has led, in recent years, to the development of the idea of “shared decision making,” namely, that patients and providers are most productively seen as partners.¹¹ If this is so, and there is much to recommend it in many (though not all) instances, then we need to assess the effect of a third partner—the computer.

There are two overriding areas of ethical concern here. The first is that the computer will create conceptual or interpersonal distance between provider and patient. Communicating about uncertainty, especially when the stakes are high, has long been a challenge for clinicians. That a computer might be used to (help) render a diagnosis causes us to run the risk of what we will call the “computational fallacy.” This is the view that what comes out of a computer is somehow more valid, accurate, or reliable than human output. Providers and patients who take such a view introduce a potentially erosive, if not destructive, element into shared decision-making contexts. Anything that increases the likelihood that a patient decision or choice will be perceived as misguided or stupid adds to the problem that shared decision making was supposed to solve.

Now, it might be supposed that the physician or nurse can eliminate at least some of this tension by not disclosing to a patient that decision support software was used in his or her case. But this introduces our second area of ethical concern, namely, the question whether patients should be given this information. The answer to this question must be determined against a background shaped by: (1) patient sophistication and understanding of medical and statistical information, and (2) clinician sophistication and understanding of communication approaches and strategies. In any case, it is inappropriate to use computer data or inferences to trump hesitant patients, or bully them into agreeing with a health professional.¹²

This point has been made most clearly in the discussion of prognostic scoring systems, or software used in critical care medicine in part to predict patient mortality. On the one hand, patients with poor prognoses might still benefit from extensive interventions, and these benefits might be important enough for the patient and/or family to seek them; on the other hand, patients with good survival odds might judge the prolongation of life to be of little value when weighed against the difficulty or burden of extensive interventions.¹³

A related issue is likely to arise with increased frequency as patients gain access to decision support software and use it to make demands on physicians, or at least to challenge or second-guess them. The difficulties raised by these demands and challenges will multiply as these systems improve. As discussed in Chapter 11, there is a sense in which one might regard such access as an important tool in the process of shared decision

making: it will not do to expect patients to become involved in their own care and simultaneously constrain their sources of information. Contrarily, a patient might constitute a paradigm case of an inappropriate decision support system user, especially in those cases in which the system causes someone to forgo appropriate medical care.

We might compare patient use of clinical decision support systems to patient use of medical texts and journals. In years past, there was an inclination to regard such access as risky and hence inappropriate. While a little knowledge can be dangerous, a position that does not go beyond such a view seems to miss an opportunity to educate patients about their illnesses and the relation between medical literature on the one hand, and medical knowledge and practice on the other. Much the same point can be made about patient use of diagnostic tools: a physician should respond to such use by making clear that computers are not surrogates for health professionals and that the practice of medicine or nursing entails far more than statistical induction from signs, symptoms, and lab values. To be sure, it would be well if actual practice embodied this insight.

As long as the healing professions are practiced in a matrix of scientific uncertainty and patient values, we err if we appoint computational decision support as a surrogate for compassionate communication, shared decisions, and quality care by competent humans.

Legal and Regulatory Issues

Computers and software raise conceptually fascinating and important practical questions about responsibility and liability. Further, the question of whether a decision-support system is a medical device needing governmental regulation is a source of tension and debate. In both domains, scientists, clinicians, philosophers, lawyers, and government and policy officials must grapple with a variety of knotty problems.

The intersection of medicine, computational decision support, and law, has been addressed mostly in speculative terms. The use of CDSS is not widespread enough to have stimulated legislation or illuminating precedent. Moreover, medicine and computing share little in the way of a common legal history. The following observation is as apt today as it was more than twenty years ago:

“The introduction of computerized decision making will require the merger of computer science and medical care; two areas with fundamentally different legal traditions. The legal differences between the computer field and medicine are striking. Medicine is tightly regulated at all levels. Most health-care providers are licensed, and a rigid hierarchical system is the norm. Yet computer systems and companies are created in a totally unregulated competitive environment in which “software piracy” is common, standardization is in its infancy, licensing is a method of transferring trade secret software, and companies begin in garages.”¹⁴

Liability and Decision Support

The overriding legal issue related to computational decision support is liability for use, misuse, or even lack of use of a computer to make or assist in rendering medical decisions.¹⁵⁻¹⁸ In the United States, tort law holds providers of goods and services accountable for injuries sustained by users. Because of legal and regulatory variation, there are similarities and differences in other countries.¹⁹⁻²¹ Such accountability is addressed by either the negligence standard or the strict liability standard.

The negligence standard applies to services, and strict liability applies to goods or products, although negligence can sometimes also apply to goods, as in cases of negligent product design. There is no consensus about whether decision-support systems are services or products, in part because these systems have properties that resemble both services and products.^{2,14-15,22-23} For instance, a physician's diagnosis is clearly a service, and any liability for erroneous diagnoses is judged by the negligence standard. If a human diagnosis is considered a service, then, it is argued, a computer diagnosis (or the task of writing the computer code that rendered the diagnosis) should have the same status. Contrarily, commercial CDSS are manufactured, mass-marketed, and sold like entities uncontroversially regarded to be products.

An additional complication is that these systems are sold to hospitals, physicians, patients, and others, and, indeed, are now available on the World Wide Web. If a patient is injured by a defective system, it remains to be determined who used the system (the physician? the patient?) and whether it was misused. Also, it can be exquisitely difficult to identify the defect in a computer program,¹⁵ as well as to answer the important question as to whether a physician could have intervened and prevented the application of mistaken advice.²

Neither is there a clear standard of care for use of decision-support software by clinicians. Physicians or nurses might someday be found negligent either for accepting a mistaken computer diagnosis or, having erred in diagnosis themselves, for failing to have used a decision-support system that might have proved corrective. In either case, the determination of negligence will have to be weighed against prevailing community or reasonable-person standards. As with other areas of practice, errors will increase liability accordingly as the practitioner is seen to have fallen behind, or moved too far ahead of, such standards.

There is a clear need for additional conceptual analysis to assist the law in sorting out these puzzles. Local trial courts and juries will often be out of their depth if called on to adjudicate liability claims that challenge fundamental conceptions of responsibility, accountability, and blame. Similar difficulties arise in other areas, such as in the intellectual property arena, when there is a need to determine whether computer software is an invention or a work of art. In one interesting approach to these questions, Prof. John Snapper attempts an account of responsibility that will not impede the

future—and presumably salutary—development of mechanical decision support. On this account, the attribution of responsibility and duty to computers for certain actions will maximize the good that will result from increased use of improved decision-support systems.²⁴ The idea is that use of conceptually inadequate legal tools to punish system designers, owners, and users, might have a chilling effect on the evolution of decision-support technology. Spreading responsibility around, and including computers as agents to which responsibility may be assigned, is said to offer the potential of stimulating system design and the benefits this would entail.

This much is clear: physicians and nurses who revile and disdain computers will be ignorant of machines that can, in principle, improve their practice and, hence, patient care. Zealots who take computers to constitute adequate or even superior human surrogates will have lost touch with the human foundations of their profession. At either extreme, the risk is high of falling outside emerging standards. This is a mistake—in ethics and at law.

Regulation of Decision-Support Software

While the history of governmental regulation of healthcare products is traceable to the Pure Food and Drug Acts of 1906, the regulation of medical devices was not formalized until the Federal Food, Drug, and Cosmetic Act of 1938. There, medical devices were defined as “instruments, apparatus, and contrivances, including their components, parts, and accessories intended: (1) for use in diagnosis, cure, mitigation, treatment, or prevention of diseases in man or other animals; or (2) to affect the structure or any function of the body of man or other animals.”^{25–26} In 1976, motivated by the increased complexity of devices and by reports of some devices’ shortcomings and failures, Congress approved comprehensive Medical Device Amendments to the 1938 regulations; the amendments were to “ensure that new devices were safe and effective before they were marketed.”^{27–28} In 1990, a new regulation replaced that emphasis on premarket approvals with an emphasis on postmarket surveillance.²⁹ Proposals to regulate diagnostic software have been evaluated against the 1976 and 1990 laws and a broad array of draft policies and statements.

The U.S. Food and Drug Administration (FDA) unequivocally regards medical software as a device. The FDA identifies four types of devices:

1. Educational and Bibliographic Software

Federal authorities regard the following as exempt from, or not falling under, existing regulation:

- Software intended only for use in performing traditional “library” functions, such as storage, retrieval, and dissemination of medical information (i.e., functions that are traditionally carried out using medical textbooks and journals).

- Software intended only for use as general accounting or communications functions.
- Software solely intended for educational purposes, rather than to diagnose or treat patients.³⁰

2. Software Components

Some software is incorporated into medical devices and is actively regulated. Examples include the software in:

- infusion pumps;
- pacemakers;
- ventilators;
- magnetic resonance imaging devices;
- diagnostic X-ray systems;
- clinical laboratory instruments;
- blood grouping instruments.³⁰

3. Software Accessories

Software accessories are attached to, or used with, other devices, and as such are also actively regulated. These include software for:

- radiation treatment planning;
- conversion of pacemaker telemetry data;
- conversion, transmission or storage of medical images;
- off-line analysis of EEG data;
- digital analysis and graphical presentation of EEG data;
- calculation of rate response for a cardiac pacemaker;
- perfusion calculations for cardiopulmonary bypass;
- calculation of bone fracture risk from bone densitometry data;
- statistical analysis of pulse oximetry data;
- calculation of refractive power of intraocular lenses.³⁰

4. Stand-Alone Software

The most controversial class, stand-alone software, includes CDSS and other decision support systems. Whether or how such systems should be regulated is a matter of continuing debate. Examples include:

- blood bank software systems which control donor deferrals and release of blood products;
- software designed to assist a healthcare practitioner in arriving at a diagnosis of a particular patient;
- software which analyzes for potential therapeutic interventions for a particular patient;
- software which records medical information for later recall, analysis, or action by a healthcare practitioner (e.g., hospital information systems, prescription ordering, drug interaction information systems, emergency room triage software, and various calculators which automate calculations of drug doses).³⁰

In 1989, an FDA draft policy proposed regulatory exemption for “Previously unclassified information management products . . . such as expert or

knowledge-based systems, artificial intelligence, and other types of decision-support systems intended to involve competent human intervention before any impact on human health occurs.”³¹ The question then became whether CDSS were intended to involve competent human intervention. This remains an interesting and important policy—and conceptual—issue. In Chapter 5, Miller and Geissbuhler examine some of the issues connected with FDA regulation.

While the FDA regards software as a device, there are a number of reasons why it might be best if medical decision-support software were not subjected to thorough federal regulation. The most common arguments against regulation include the following:

- Software is most accurately regarded as a mental construct or abstract entity, i.e., the sort of thing not customarily falling within the FDA’s regulatory purview.
- Practitioners—not software—have traditionally been subjected to licensing requirements.
- Software evolves rapidly and locally, and any sort of national software monitoring is likely to be ineffective or impossible.
- Software is imperfect, and so improvement and refinement—not perfection—must be the standard to be striven for and met. Yet at law, strict liability standards (usually applied to devices or goods but not services) require perfection.

Several of these points could be in line with an influential stance held by a former commissioner of the agency, namely that the FDA should “apply the least regulation allowed to remain consistent with the requirements of public health and safety.”³²

The debate over medical software regulation represents one of the most important controversies of the Computer Age. The balancing of risks and benefits, as well as public safety and technological progress, means that scientists, clinicians, and policy makers have one of civilization’s most interesting—and challenging—tasks.

Conclusion and Future Directions

Clinicians, philosophers, lawyers, and policy makers have grappled for more than a decade with social, ethical, and legal issues raised by the growth of health informatics, perhaps especially by progress in development of tools for clinical decision support. What has emerged is a recognition that future scientific growth must be guided by corresponding attention to ethical issues. These issues address the role of: error avoidance and standards; appropriate use and users; and professional relationships. Scientific programs and publications may be regarded as duty-bound to foster environments in which further attention to ethical, legal, and social issues is

encouraged. Indeed, to the extent that morality guides the law, vigorous programs to identify and debate ethical issues will be of no small service to society as legislatures, courts, and government regulators and policy makers attempt to apply the insights of ethics to practical problems in health informatics.

More research on ethical issues involved in use of CDSS is essential for this process. We have, for instance, only begun to address issues that arise when diagnostic tools are made available on the World Wide Web. We are in no way clear about the level of ethics education that is appropriate for students in health informatics, and there is much work to be done at the intersections of ethics and system evaluation, and of ethics and standards of care.

Elsewhere in the history of science and technology, such challenges are often taken to constitute evidence of the growth and maturation of an applied science. This is no less true for clinical decision support systems and, indeed, for all of health informatics.

References

1. Hippocrates. Prognosis. In Lloyd GER, ed. Hippocratic writings, translated by Chadwick J, Mann WN. London: Penguin Books; 1983:170–185.
2. Miller RA, Schaffner KF, Meisel A. Ethical and legal issues related to the use of computer programs in clinical medicine. *Ann Intern Med* 1985;102:529–536.
3. de Dombal FT. Ethical considerations concerning computers in medicine in the 1980s. *J Med Ethics* 1987;13:179–184.
4. Miller RA. Why the standard view is standard: people, not machines, understand patients' problems. *J Med Philos* 1990;15:581–591.
5. Goodman KW, ed. Ethics, computing and medicine: informatics and the transformation of health care. Cambridge and New York: Cambridge University Press; 1997.
6. Miller RA, Goodman KW. Ethical challenges in the use of decision-support software in clinical practice. In: Goodman KW, ed. Ethics, computing and medicine: informatics and the transformation of health care. Cambridge and New York: Cambridge University Press; 1997:102–115.
7. Berner ES, Webster GD, Shugerman AA, et al. Performance of four computer-based diagnostic systems. *N Engl J Med* 1994;330:1792–1796.
8. Sorace JM, Berman JJ, Carnahan GE, Moore GW. PRELOG: precedence logic inference software for blood donor deferral. *Proc Annu Symp Comput Appl Med Care* 1991:976–977.
9. Boon ME, Kok LP. Neural network processing can provide means to catch errors that slip through human screening of Pap smears. *Diag Cytopathol* 1993;9:411–416.
10. Goodman KW. Bioethics and health informatics: an introduction. In: Goodman KW, ed. Ethics, computing and medicine: informatics and the transformation of health care. Cambridge and New York: Cambridge University Press; 1997:1–31.
11. Forrow L, Wartman SA, Brock DW. Science, ethics, and the making of clinical decisions. *JAMA* 1988;259:3161–3167.

12. Goodman KW. Outcomes, futility, and health policy research. In: Goodman KW, ed. *Ethics, computing and medicine: informatics and the transformation of health care*. Cambridge and New York: Cambridge University Press; 1997:116–138.
13. Brody BA. The ethics of using ICU scoring systems in individual patient management. *Prob Crit Care* 1989;3:662–670.
14. Brannigan VM, Dayhoff RE. Medical informatics: the revolution in law, technology, and medicine. *J Leg Med* 1986;7:1–53.
15. Miller RA. Legal issues related to medical decision-support systems. *Int J Clin Monit Comput* 1989;6:75–80.
16. Mortimer H. Computer-aided medicine: present and future issues of liability. *Computer Law J* 1989;9:177–203.
17. Turley TM. Expert software systems: the legal implications. *Computer Law J* 1988;8:455–477.
18. Hafner AW, Filipowicz AB, Whitely WP. Computers in medicine: liability issues for physicians. *Int J Clin Monit Comput* 1989;6:185–194.
19. Beier B. Liability and responsibility for clinical software in the Federal Republic of Germany. *Comput Methods Programs Biomed* 1987;25:237–242.
20. Brahams D, Wyatt J. Decision aids and the law. *Lancet* 1989;ii:632–634.
21. Allaert FA, Dussere L. Decision support system and medical liability. *Proc Annu Symp Comput Appl Med Care* 1992:750–753.
22. Birnbaum LN. Strict products liability and computer software. *Computer Law J* 1988;8:135–156.
23. Gill CJ. Medical expert systems: grappling with the issues of liability. *High Technol Law J* 1987;1:483–520.
24. Snapper JW. Responsibility for computer-based decisions in health care. In: Goodman KW, ed. *Ethics, computing and medicine: informatics and the transformation of health care*. Cambridge and New York: Cambridge University Press; 1997:43–56.
25. Munsey RR. Trends and events in FDA regulation of medical devices over the last fifty years. *Food and Drug Law J* 1995;50:163–177.
26. Public Law No. 75-717, 52 Stat. 1040 (1938), as amended 21 U.S.C. Sections 301 et seq (1988).
27. Kessler DA, Pape SM, Sundwall DN. The federal regulation of medical devices. *N Engl J Med* 1987;317:357–366.
28. Public Law No. 94-295, 90 Stat. 539 (1976), codified at 21 U.S.C. Sections 360c et seq (1982).
29. Brannigan VM. Software quality regulation under the Safe Medical Devices Act of 1990: hospitals are now the canaries in the software mine. *Proc Annu Symp Comput Appl Med Care* 1991:238–242.
30. Food and Drug Administration. FDA regulation of medical device software. (Document prepared for an FDA Software Policy Workshop, Sept. 3–4, 1996, National Institutes of Health, Bethesda, Md.). <http://www.fda.gov/cdrh/ost/points.html>.
31. Food and Drug Administration, Center for Devices and Radiological Health. Policy for the regulation of computer products, draft, 13 November 1989. Rockville, Maryland: FDA, CDRH; 1989.
32. Young FE. Validation of medical software: present policy of the Food and Drug Administration. *Ann Intern Med* 1987;106:628–629.

7

Clinical Trials of Information Interventions

E. ANDREW BALAS and SUZANNE AUSTIN BOREN

When a clinical decision support system (CDSS) passes the test of accuracy and is ready for clinical implementation, the need for replicable and generalizable measurement of practical impact emerges. It is increasingly acknowledged that measurement of system performance and impact represents the research component of informatics projects, and that such evaluations should guide the development of decision support technologies.^{1,2} This chapter discusses the methodology for systematic evaluation of information interventions. It provides a framework for designing appropriate tests of the clinical impact of CDSS.

Several studies have demonstrated that computers are able to influence the behavior of providers, management of patients, and outcome of health care in many clinical areas.³⁻⁸ Unfortunately, claims for computerized medical information systems seem to exceed the documented benefits. Many predictions about the computer revolution have not been realized, and the evidence arising from various clinical experiments is often controversial.⁹⁻¹² There is an increasing demand to provide convincing evidence of the benefits of clinical information services.¹³⁻¹⁵

The Practical and Scientific Need for Clinical Testing

Few medical questions have been more controversial than the clinical usefulness of computer systems. Early on in the development of clinical computing applications, it was suggested that the ability of computers to store information on patient history, physical findings, and laboratory data would assist in decision making, thereby freeing the physician to focus on other aspects of clinical care.¹⁶ However, enthusiasm for the potential of the computer as an intellectual tool eroded quickly. For example, some studies indicated that a computer system for diagnosing abdominal pain generated more accurate information and reduced perforation rate.^{17,18} Other studies concluded that the same system had no useful role in this diagnosis.^{19,20}

Early computer system evaluations often assumed that more patient information meant better patient care. However, evaluation of techniques such as electronic fetal heart rate monitoring illustrate that this is not always the case. In the early 1970s, the common perception was that continuous heart rate monitoring can protect the fetus from prolonged intrauterine oxygen deprivation.^{21,22} Subsequently, several controlled clinical trials failed to demonstrate any clinical benefit of this technology.²³⁻²⁵

Evaluators of clinical computer applications have repeatedly criticized insufficient demonstration of quality improvement. In a review of reports on clinical computer systems, over 75% of 135 articles were anecdotal, and only half of the remainder met basic scientific criteria for the conduct of clinical trials.¹³ Piantadosi and Byar¹⁴ concluded that a basic shift is required in how scientists view research concepts as opposed to research results; the former are generally not considered proper objects for review or dissemination. Similar issues have been raised in other areas of health sciences. For example, Tyson et al.²⁶ conducted a review of therapeutic studies in perinatal medicine and found only 10% of the reports presented conclusions of the investigators that were supported by the evidence they presented.

Some argue that medical information systems need not justify themselves in terms of improved patient outcomes because these systems are designed to influence primarily the providers of health care.²⁷ Therefore, only the change in the process of care has to be demonstrated (e.g., performance of clinicians). This argument is acceptable when the process of care affected has an obvious relationship to healthcare outcomes (e.g., certain cancer screening procedures). However, there are numerous aspects of health care for which the relationship between process and outcome is unclear (e.g., completeness of medical records).

Nevertheless, in order to compete for the resources of healthcare providers, system developers have to demonstrate the relevance of their computer programs to healthcare quality improvement and cost control.²⁸ Medical practice involves a tremendous amount of information processing: collecting patient data, sharing information with patients, decision making in diagnostics and therapeutics, documenting care, communicating with other healthcare professionals, and educating patients. Healthcare organizations invest on average only 2.6% of their operating budget in information technology, a marked contrast with the average 8% to 9% invested by the banking industry.²⁹ During the past decades, computer systems have become active ingredients of health services, but the assessment of the new information technology is still considered to be a controversial issue. Practitioners interested in applying the new technologies need information on the results of the clinical evaluation of computer systems.

The recurrent debate over healthcare reform and the intensive search for cost-effective methods to improve patient care, repeatedly highlight the need for adequate technology assessment of clinical information systems.

Although early evaluation studies focused on the accuracy of information generated by the computer system, newer studies tend to focus on differences in the process or outcome of care due to the computer system. Although health care is clearly an information-intensive service, the clinical value of computer applications is often questioned due to the lack of demonstrated clinical benefits. As healthcare organizations are actively searching for opportunities to improve their information systems through purchase or development, the example set by systems on the market is very important for practical and theoretical purposes as well.

Research Methods to Demonstrate Practical Impact

There is a growing demand for adequate technology assessment in the field of medical informatics.^{13,14,30,31} Medical technology includes drugs, devices, and procedures used in medical care, as well as the organizational and supportive systems that provide such care. Technology assessment provides practitioners with information on alternative techniques. The pioneering report of Cochrane noted that many standard medical practices lack evidence of effectiveness.³² Concerns of costs also stimulate efforts to assess the practical value of not only new, but also established, technologies. Some argue that the assessment of healthcare technologies should be an iterative process and that there is a need to continuously reassess existing technologies by combining evidence from all reliable sources.^{32,33}

As Berwick notes, Deming's theory of continuous quality improvement depends on understanding and revising the production processes on the basis of data about the processes themselves.³⁴ Likewise, quality improvement efforts in health care depend on measurable quality objectives and appropriate interventions and changes in the process. Particularly, randomized controlled trials (RCTs) have direct relevance to healthcare quality improvement as they become increasingly important sources of information about the clinical value of various interventions (e.g., physician and patient education,³⁵ interventions to promote cancer screening,³⁶ computerized medical records,³⁷ and home care after hospital discharge³⁸).

The concept of demonstrating quality improvement by measurements is accepted in the field of medical informatics. Clinical computer system designers often use benchmark tests, surveys, and historical control comparisons to indicate the quality improvement resulting from the use of the new system. However, benchmark tests only measure the technical performance of the computer programs. They do not provide useful data on the impact of the system on either the process or outcomes of care. On the other hand, surveys of users' opinions only provide indirect information about the difference the system made in patient care.

Comparison with historical controls (before-after study) is a popular method of evaluating clinical computer applications. The fact that computer

systems are often connected to a patient database further encourages the use of historical controls as a baseline for evaluation.³⁹ Although they may provide some useful information, analyses of databases or historical control groups of patients cannot replace planned clinical experimentation.⁴⁰ The greatest concern in using historical controls is that there may be a confounding bias introduced by the different time periods. Definitions of disease and diagnostic testing methods may change over time. In the database, data may be missing either because they were lost or not recorded. Furthermore, developing hypotheses after the collection of data often leads to unplanned multiple comparisons.⁴¹ Excessive numbers of statistical tests can easily result in misleading statistical significance, but no practical significance.

Randomized controlled clinical studies can provide the most valid information about the efficacy of computerized information systems in patient care.⁴² From 1985–1995, the number of randomized controlled clinical trials testing computerized information interventions increased an average of 50% annually.⁴²

A review of clinical trials of clinical decision support systems provides strong evidence that some clinical decision support systems can improve physician performance.^{43,44} However, the majority of studies assessing patient outcomes did not demonstrate significant improvements. In addition, there have been very few controlled studies of CDSS, which have a diagnostic, as opposed to a therapy focus.

User Satisfaction with Decision Support Systems

Measuring and managing users' attitudes toward various aspects of information systems is an important part of making computer systems successful. No clinical computer system can be successful without gaining the support of practitioners. The primary challenge of measurement is to find an appropriate control for comparison. Ideally, satisfaction should be measured before and after the introduction of the new decision-support system, and there should be an improvement in users' satisfaction. However, it is often challenging to develop a generic user-satisfaction instrument.

There are many complex beliefs, attitudes, and behaviors influencing computer use among healthcare professionals. A critical success criterion for how useful information systems are, is the way in which computer users react to various aspects of the system. If overall satisfaction levels are high, the user will adapt his/her activities to take advantage of the computer. The user may not cooperate and may become antagonistic toward the system if satisfaction is too low. Questionnaires or surveys are tools that can be used to assess user attitudes. The particular significance of surveys is their ability to measure the acceptance of the system and the satisfaction of the users. However, a system must be used appropriately before its impact can be

accurately measured. Inattention on the part of system developers to the specific clinical needs of end users may result in system underutilization or sabotage.^{45,46}

Teach and Shortliffe⁴⁷ found that physician attitudes regarding computer-based clinical decision aids and a medical computing tutorial were generally favorable. Physician expectations about the effect of computer-assisted consultation systems on medical practice were also positive, although there were considerable differences among physicians. In addition, the tutorial produced a substantial increase in knowledge about computing concepts and a significant effect on physician demands.

Decision-support modules built into the Health Evaluation through Logical Processes (HELP) system are described in more detail in Chapter 8. HELP is a clinical information system developed at LDS Hospital that includes a computer-based patient record, alerts, reminders, and other decision-support aids. Gardner and Lundsgaarde measured the attitudes of physicians and nurses who used the HELP system through a questionnaire with fixed-choice questions supplemented with free-text comments.⁴⁸ The respondents did not feel that computerized decision support decreased their decision-making power, nor did they feel that expert computer systems would compromise patient privacy or lead to external monitoring. The results of the survey indicated that experience with a system was the best way to break down attitudinal barriers to the use of that system. Although surveys and questionnaires can provide direct evidence of user attitudes toward CDSS, they are only an indirect measure of the behavioral impact of these systems.

Randomized Controlled Clinical Trials of Decision-Support Services

Because medical practice requires the efficient management of information, providing information to physicians is increasingly recognized as a clinical intervention designed to influence the process and/or outcome of patient care.^{40,49} The quality of care is expected to be improved by the advanced methods of decision support. However, the benefits have to be demonstrated by appropriately controlled clinical measurements. There are many types of randomized clinical trials (e.g., parallel designs, factorial designs, cross-over trials), but the basic principles are the same: prospective and contemporaneous monitoring of the effect of a randomly allocated intervention. It is widely accepted that clinical trials represent a design superior to before-and-after studies (vulnerable to changes, over time, that are unrelated to the effect of the intervention) or matched control studies (a much less reliable method of obtaining comparable groups of subjects). Today, drugs, surgical procedures, alternative care delivery techniques, and

computerized decision-support services are evaluated in randomized controlled trials. For example, Pozen et al.⁵⁰ tested a predictive instrument to reduce admissions to the coronary care unit. They found that the instrument had the potential to reduce coronary care unit admissions by 250,000 for acute ischemic heart disease.

As necessary as RCTs are, they also have limitations. RCTs can test only specific hypotheses about selected aspects of computer systems. For instance, no single RCT can answer the question as to whether an integrated hospital system is good or bad. Selected information systems can be good for certain types of patients, indifferent for others, and only potentially useful for a third group of patients. Experimental evaluations of clinical computer applications (computer-assisted services) need to identify the specific conditions to be treated, specific interventions to be tested, and specific outcome variables to be measured. If this is done, the results can be specific, interpretable, and useful for practical purposes.

A surprisingly high proportion of trials are performed in outpatient facilities, particularly in primary care, while relatively few trials evaluated hospital information systems. This finding is in contrast to the large sums of money spent on information systems for inpatient care.

Although clinical trials are rapidly gaining acceptance in technology assessment, the methodology of such trials does not seem to be common knowledge. Several techniques commonly used in drug trials are irrelevant in testing computerized information interventions (e.g., blinding to the intervention, placebo), while other aspects are more critical (e.g., detailed description of sites, technical specification of intervention). The evaluated effect can be either a change in the process of care (e.g., increased or reduced use of certain drugs) or in the outcome of care (e.g., lower rate of infections). A particular weakness of many trials of computer systems is the lack of evaluation of patient outcome. It is certainly understandable that many information service trials evaluate the effect on care processes, since their main intent is to influence the process through the provision of accurate and timely information. However, documenting decreased side effects or other outcome measures, such as lower complication rates, could probably convince more clinicians as to their usefulness.

The setting in which the trial is conducted is critical to the representativeness of the trial. For example, the guidelines of the Nordic Council on Medicines recommend that the selection of a site for the trial has to be dependent on the potential risks involved to ensure satisfactory safety for the subjects.⁵¹ It is a reasonable expectation that the site of a trial should represent the actual settings where the intervention will ordinarily be applied, otherwise, the generalization of the results are questionable. Many RCTs have tested the effect of various interventions on the practice patterns of residents in large academic centers. It is frequently assumed that the effects will be identical when board certified physicians are subjected

to the same intervention in a nonacademic environment, a hypothesis which has never been evaluated.

In health services research, randomization often assigns patients to groups through their healthcare providers. Major textbooks on clinical trials describe a large variety of randomization techniques.⁵² The common feature of these techniques is that the patient is the unit of randomization. In health services research, it is often the provider who is directly targeted by the intervention. Therefore, the provider should be the unit of randomization and patients or encounters are randomized only through their providers. Our studies documented that only one-third of the trials on computer systems used an appropriate randomization technique.⁵³ The use of provider as a unit of randomization works well and could be more widely used in health services research. However, the number of providers has to be sufficient to ensure representativeness of not only the patient sample, but also of the provider sample. It is difficult to accept trials that randomize through a small number of provider units (e.g., patients of one hospital are in the study group while patients of another hospital are in the control group). In most cases, trials that randomize through less than six provider units should not be accepted as valid sources of evidence.

Columbia Registry of Medical Management Trials

Improving quality of care is not only a professional and ethical concern of physicians, but also the most important challenge facing a healthcare organization today.³⁴ Advanced computer techniques promise significant improvement in the quality of care through increased use of appropriate procedures and reduced use of unnecessary and potentially harmful procedures. Cochrane⁵⁴ emphasized the need to summarize evidence derived from randomized controlled trials as distinct from other kinds of evidence and to organize critical summaries by specialty or subspecialty of all relevant randomized controlled trials.

Various trial registries have been established in an attempt to improve access to published reports. Many of these registries deal with perinatal care, management for AIDS, or cancer treatment (e.g., the Oxford Perinatal Database,⁵⁵ the AIDS Clinical Trials Information Service, and the National Cancer Institute (NCI) Cancer Control Intervention Studies⁵⁶). Some review papers contain valuable bibliographies of clinical trials.⁵⁷ However, clinical trials testing medical management interventions, a broad area critical to health-care quality improvement and cost control, have not been the focus of any known registry.

The purpose of organizing the Columbia Registry of Medical Management Trials is to support practitioners and researchers with the best available controlled evidence on the practical value of clinical interventions changing the delivery of health services. The registry is used to facilitate

access through improved MEDLINE indexing, to develop meta-analyses and reviews, and to analyze the trial methodology in health services research. Examples of the interventions within the scope of our registry include patient education, reminders/prompts, feedback, computer-aided diagnosis-making, and computerized records. There are approximately 1,800 reports on randomized controlled trials in the registry.

Specific eligibility criteria have been developed for inclusion/exclusion of reports in the Columbia Registry of Medical Management Trials. The design of the report is the first aspect evaluated. The study must be a prospective, contemporaneously controlled clinical trial with random assignment of intervention. Trials using allocation systems similar to a random number table (e.g., alternating encounters, alternating days of the week) are also eligible. Reports that do not meet this basic criterion (e.g., nonrandomized trial groups, review articles) are not included in the registry. Second, there should be an information management intervention in the study group with no similar intervention in the control group. Often, the control group simply receives the current standard of care, as compared with the experimental intervention used in the trial. The third criterion is that the effect of the intervention on the process and/or outcome of patient care must be measured. Planned or ongoing trials are not included in the registry because they do not meet this criterion.

The Columbia Registry of Medical Management Trials serves as a valuable resource for information system developers and practitioners by systematically collecting and rearranging the knowledge from these trials into a format that can be used by practitioners and others making healthcare decisions. This knowledge engineering is accomplished in several steps. First, the trials are located by using a systematic approach to search MEDLINE, which is likely to outperform conventional searches. Each search consists of a study design concept and an intervention or effect concept. The study design concept is the same for each search and includes the following terms: random (truncated textword), group (truncated textword), random allocation (textword and MeSH), randomized controlled trial (publication type) and clinical trial (publication type). The intervention or effect concept changes depending on specific interventions or effects. Subsequently, critical information is abstracted from the registered trials, and the practical messages of such studies are made available to those who need them. The same executive summary can be used to implement organizational changes, further healthcare quality improvement, conduct meta-analyses, or write literature reviews.

Several studies documented that, regardless of the complexity of the search process, some eligible reports will remain unretrieved. Therefore, clinical trial registries grow not only through the inclusion of new publications, but also through the discovery of eligible studies published earlier. The developers of the Oxford Perinatal Database also noted that there is no "gold standard" available to judge the completeness of a registry.⁵⁵

TABLE 7.1. Information intervention categories.

Information Intervention Categories	Number of Reports (% Positive)
Patient Focus	
Computer-assisted interactive patient education, instruction and therapy	19 (74)
Patient prompt/reminder	15 (80)
Patient-computer interactive information gathering	2 (100)
Provider Focus	
Provider prompt/reminder	19 (100)
Computer-assisted treatment planner	19 (79)
Provider feedback	19 (68)
Computerized medical record and information access	19 (74)
Prediction	6 (83)
Computer-assisted diagnosis	4 (50)
Total*	98 (85)

* Some reports test several interventions.

The synthesis of trial results helps the identification of most effective information services. Table 7.1 shows the percentages of positive trials for different types of information interventions that are included in the registry.

The number of randomized controlled trials as the ultimate evidence on the practical difference made by a specific intervention is rapidly expanding. Meta-analysis is the use of statistical techniques to integrate results of separate, but similar, clinical trials. Instead of providing a qualitative assessment of a few studies, meta-analysis promises a systematic and quantitative synthesis of all available studies. Systematic collection procedures are designed to avoid the well known deficiencies of the conventional “pick-and-choose” approach.⁵⁸

Research synthesis of evidence from several randomized controlled clinical trials always raises the question of clinical efficacy. Vote-counting is an established method of expressing the success rate of a particular intervention.⁵⁹ When the number of successful trials is very high in a particular category, then the intervention is likely to make a difference. The particular advantage of vote-counting is that information on the success or failure of the intervention is available from virtually all trial reports. Obviously, vote-counting does not consider the magnitude of effect. Primary research reports not providing enough information to calculate effect size estimates usually contain information about the direction of the effect. On the other hand, meta-analyses using the popular odds-ratio methods can specify the magnitude of the effect, and are likely to discover additional categories of effective interventions.

Diversity, a frequent concern in research synthesis, can be an advantage as well as a disadvantage. Trials pooled together are always somewhat

different in their sites, samples, interventions, and effect variables. A diversity of sites and samples (within the stated pooling criteria) can help document an intervention's success under a variety of circumstances. Diverse interventions can also help to reflect the natural variability of use in different healthcare organizations. For example, it would be unreasonable to demand separate testing of physician reminders for every single clinical procedure. Successfully applying a particular information intervention in a variety of settings and disease conditions increases the generalizability of results and the intervention's practical value.

As discussed in Chapter 2, computerized decision support requires representation of clinical knowledge in Boolean production rules or other tightly organized structures (e.g., expression in probabilities, knowledge frames). To represent the data from clinical trials, into a form that can be used in CDSS, requires knowledge engineering, and the structuring of such evidence is becoming an important trend in knowledge engineering. As the amount of published scientific evidence grows, finding the right report is no longer sufficient. The report has to be supplemented with the abstraction of the specific information to meet the needs of clinicians, researchers, and policy makers. Conventional abstracts by the investigators provide useful synopses, but often lack detail and standardization. An analysis of 150 trial reports led to the development and validation of a quality scoring system which can be used as an itemized checklist to portray the methodological quality of health services research trials.⁵³

Effective Information Interventions

Randomized controlled trials confirm that four generic information interventions that are active components of computer systems can make a significant difference in patient care (patient education, treatment planning, physician and patient reminders).⁶⁰ To manage care and improve quality, computer systems of primary care should incorporate these effective information services.

Interactive patient education can help patients improve their health through health promotion, educational information on the management of medical conditions, and computerized instruction. Seventy-four percent of the patient education studies were successful. Chapter 11 includes descriptions of some of these patient education studies.

A large number of studies employed the use of computer algorithms to assist in drug dosing decision making (e.g., aminoglycoside,⁶¹ insulin,⁶² digoxin,³ phenytoin,⁶³ sodium nitroprusside,⁶⁴ lidocaine,⁶⁵ propranol,⁶⁶ and amitriptyline⁶⁷). For example, the first known trial of a decision-support system compared the effect of computed digoxin dosage to that of unaided physician judgment.³ The results indicated that the computer slightly outperformed the physician and that the correlation between predicted and measured serum digoxin concentrations was closer in the computer-assisted

patient group. Overall, 79% of the computer-assisted treatment planner studies were successful.

Reminders represent one of the primary techniques of delivering messages generated by clinical decision support systems. Reminder messages recommend specific action at the time of decision-making. Computers can scan each patient's record to identify tests and other procedures that are due. The main function of the computer system is the identification of eligible patients and triggering the use of a particular clinical procedure.

Several controlled experiments have demonstrated that physicians respond to computer-generated reminders by performing the recommended interventions (e.g., influenza immunization, mammography). For example, patients of physicians who received reminders on the encounter forms were significantly more likely to have a mammogram ordered for them.⁶⁷ Procedures frequently targeted by the provider prompt/reminder trials included cancer screening^{36,68} (stool occult blood, sigmoidoscopy, rectal examination, mammography, breast examination, Papanicolaou test, pelvic examination) and vaccinations (influenza,⁶⁹ pneumococcal,⁷⁰ tetanus,⁷¹ and infant immunizations⁷²). All of the physician reminder studies and 80% of the patient reminder studies were successful.

The syntheses of trial results from the registry have already led to several practical and significant observations. For example, our meta-analyses of randomized controlled trials testing physician reminders concluded that this is a highly effective information intervention, but the results vary depending on the targeted clinical procedure (e.g., cancer screening versus immunization).^{73,74} These and other studies have demonstrated that computers can help to make patient care more consistent by reminding physicians to order or perform recommended procedures. Many systems show significant and beneficial impact in selected clinical areas, particularly health maintenance. In addition, 95% of the studies in our systematic review of the acceptability and effectiveness of computerized patient education interventions reported positive results.⁷⁵

Value of Noncomputerized Information Interventions

Originally, the Columbia Registry of Medical Management Trials was designed to include only trials using some form of computer intervention. Once the first 100 trials had been registered, it became clear that noncomputerized information interventions could be equally valuable.

Patient education provides an example of how noncomputerized information interventions can be effective. Educating patients about good chronic care, needs to be based on scientifically sound evidence. Patient education involves more than telling people what to do or giving them instructional material to read. The growing number of randomized clinical trials testing patient information makes the casual, ad hoc, and opinion-based

approach to patient education unacceptable. People easily slip in opinions when they are describing what should be included in the education of patients. Generalization of clinical trial results appears to be a better option than just relying on opinion. There are many education topics that are definitely useful for patients, and educators should choose them over contents that have never been shown to be beneficial.

A systematic review of 170 studies involving the education of 25,970 patients with diabetes, asthma, or congestive heart failure documents that far more clinical evidence is available on patient education beyond simply confirming that education is generally useful. Numerous successful randomized controlled trials link various educational contents and methods to improved health status, social functioning, and satisfaction.⁷⁶ This systematic review has led to the development of evidence-based patient education checklists for diabetes, asthma, and congestive heart failure. The evidence base from the randomized controlled trials of patient education could be combined with information technologies to increase access to education through new approaches. Packaging of informational messages, for easier and more effective prompting, as well as alternative delivery techniques, should be analyzed in future randomized controlled trials.

Obtaining good data is the basis for decision making about the value of diagnostic and other decision support systems. As more CDSS reach the implementation stage, RCTs of their effectiveness, as an information intervention, will be possible. Registries of RCTs will be able to provide the data needed to answer questions about the value of particular CDSS, the value of CDSS in particular settings, and the value of CDSS for particular purposes.

References

1. Friedman CP. Where's the science in medical informatics? *JAMIA* 1995;2:65–67.
2. Sullivan F. What is health informatics? *J Health Serv Res Policy* 2001;6:251–254.
3. Peck CC, Sheiner LB, Martin CM, Combs DT, Melmon KL. Computer-assisted digoxin therapy. *N Engl J Med* 1973;289:441–446.
4. McAlister NH, Covvey HD, Tong C, Lee A, Wigle ED. Randomized controlled trial of computer-assisted management of hypertension in primary care. *Br Med J* 1986;293:670–674.
5. Tierney WM, Miller ME, McDonald CJ. The effect on test ordering of informing physicians of the charges for outpatient diagnostic tests. *N Engl J Med* 1990;322:1499–1504.
6. Tamblyn R, Huang A, Perreault R, et al. The medical office of the 21st century (MOXXI): effectiveness of computerized decision-making support in reducing inappropriate prescribing in primary care. *CMAJ* 2003;169:549–556.
7. McCowan C, Neville RG, Ricketts IW, Warner FC, Hoskins G, Thomas GE. Lessons from a randomized controlled trial designed to evaluate computer decision support software to improve the management of asthma. *Med Inform Internet Med* 2001;26:191–201.

8. Dexter PR, Perkins S, Overhage JM, Maharry K, Kohler RB, McDonald CJ. A computerized reminder system to increase the use of preventive care for hospitalized patients. *N Engl J Med* 2001;345:965–970.
9. Wones RG. Failure of low-cost audits with feedback to reduce laboratory test utilization. *Med Care* 1987;25:78–82.
10. Wyatt J, Spiegelhalter D. Field trials of medical decision-aids: potential problems and solutions. *Proc Annu Symp Comput Appl Med Care* 1991:3–7.
11. Montgomery AA, Fahey T, Peters TJ, MacIntosh C, Sharp DJ. Evaluation of computer based clinical decision support system and risk chart for management of hypertension in primary care: randomised controlled trial. *BMJ* 2000;320: 686–690.
12. Hetlevik I, Holmen J, Kruger O, Kristensen P, Inversen H, Furuseth K. Implementing clinical guidelines in the treatment of diabetes mellitus in general practice. Evaluation of effort, process, and patient outcome related to implementation of a computer-based decision support system. *Int J Technol Assess Health Care* 2000;16:210–227.
13. Haynes RB, Walker CJ. Computer-aided quality assurance: a critical appraisal. *Arch Intern Med* 1987;147:1297–1301.
14. Piantadosi S, Byar DP. A proposal for registering clinical trials. *Control Clin Trials* 1988;9:82–84.
15. Dickersin K, Rennie D. Registering clinical trials. *JAMA* 2003;290:516–523.
16. Gorry GA. Computer-assisted clinical decision making. *Methods Inf Med* 1973;12:45.
17. de Dombal FT, Leaper DJ, Staniland JR, McCann AP, Horrocks JC. Computer aided diagnosis of acute abdominal pain. *Br Med J* 1972;2:9–13.
18. McAdam WA, Brock BM, Armitage T, Davenport P, Chan M, de Dombal FT. Twelve years experience of computer-aided diagnosis in a district general hospital. *Ann R Coll Surg Engl* 1990;72:140–146.
19. Sutton GC. How accurate is computer-aided diagnosis? *Lancet* 1989;2:905–908.
20. Paterson-Brown S, Vipond MN, Simms K, Gatzen C, Thompson JN, Dudley HA. Clinical decision-making and laparoscopy versus computer prediction in the management of the acute abdomen. *Br J Surg* 1989;76:1011–1113.
21. Gabert HA, Stenchever MA. Continuous electronic monitoring of fetal heart rate during labor. *Am J Obstet Gynecol* 1973;115:919–923.
22. Paul RH, Hon EH. Clinical fetal monitoring. V. Effect on perinatal outcome. *Am J Obstet Gynecol* 1974;118:529–533.
23. McDonald D, Grant A, Sheridan-Pereira M, Boylan P, Chalmers I. The Dublin randomized controlled trial of intrapartum fetal heart rate monitoring. *Am J Obstet Gynecol* 1985;152:524–539.
24. Shy KK, Luthy DA, Bennett FC, et al. Effects of electronic fetal heart-rate monitoring, as compared with periodic auscultation, on the neurologic development of premature infants. *N Engl J Med* 1990;322:588–593.
25. Freeman R. Intrapartum fetal monitoring—a disappointing story. *N Engl J Med* 1990;322:624–626.
26. Tyson JE, Furzan JA, Resich JS, Mize SG. An evaluation of the quality of therapeutic studies in perinatal medicine. *J Pediatr* 1983;102:10–13.
27. Simborg DW, Whiting-O’Keefe QE. Evaluation methodology for ambulatory care information systems. *Med Care* 1982;20:255–265.

28. Kreps GL. Evaluating new health information technologies: expanding the frontiers of health care delivery and health promotion. *Stud Health Technol Inform* 2002;80:205–212.
29. Smith L. The coming health care shakeout. *Fortune* 1993;17:70–75.
30. Kazanjian A, Green CJ. Beyond effectiveness: the evaluation of information systems using A Comprehensive Health Technology Assessment Framework. *Comput Biol Med* 2002;32:165–177.
31. Goodman CS, Ahn R. Methodological approaches of health technology assessment. *Int J Med Inform* 1999;56:97–105.
32. Cochrane AL. Effectiveness and efficiency: random reflections on health services. London: Nuffield Provincial Hospitals Trust; 1972.
33. Banta HD, Thacker SB. The case for reassessment of health care technology: once is not enough. *JAMA* 1990;264:235–240.
34. Berwick DM. Continuous improvement as an ideal in health care. *N Engl J Med* 1989;320:53–56.
35. Vinicor F, Cohen SJ, Mazzuca SA, et al. DIABEDS: a randomized trial of the effects of physician and/or patient education on diabetes patient outcomes. *J Chronic Dis* 1987;40:345–356.
36. McPhee SJ, Bird JA, Jenkins CN, Fordham D. Promoting cancer screening: a randomized, controlled trial of three interventions. *Arch Intern Med* 1989;149:1866–1872.
37. Tape TG, Campbell JR. Computerized medical records and preventive health care: success depends on many factors. *Am J Med.* 1993;94:619–625.
38. Melin AL, Bygren LO. Efficacy of the rehabilitation of elderly primary health care patients after short-stay hospital treatment. *Med Care* 1992;30:1004–1015.
39. Starmer CF, Lee KL, Harell FE, Rosati RA. On the complexity of investigating chronic illness. *Biometrics* 1980;36:333–335.
40. Byar DP. Why data bases should not replace randomized clinical trials. *Biometrics* 1980;36:337–342.
41. Ingelfinger JA, Mosteller F, Thibodeau LA, Ware JH. *Biostatistics in clinical medicine*. 2nd ed. New York: Macmillan; 1987.
42. Balas EA, Stockham MG, Mitchell MA, Austin SM, West DA, Ewigman BG. The Columbia registry of information and utilization management trials. *JAMIA* 1995;2:307–315.
43. Hunt DL, Haynes RB, Hanna SE, Smith K. Effects of computer-based clinical decision support systems on physician performance and patient outcomes: a systematic review. *JAMA* 1998;280:1339–1346.
44. Garg AX, Adhikari NK, McDonald H, et al. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. *JAMA*. 2005;293:1223–1238.
45. Williams LS. Microchips versus stethoscopes: Calgary hospital, MDs face off over controversial computer system. *CMAJ* 1992;147:1534–1547.
46. Massaro TA. Introducing physician order entry at a major academic medical center: I. Impact on organizational culture and behavior. *Acad Med* 1993;68:20–25.
47. Teach RL, Shortliffe EH. An analysis of physician attitudes regarding computer-based clinical consultation systems. *Comput Biomed Res* 1981;14:542–558.
48. Gardner RM, Lundsgaarde HP. Evaluation of user acceptance of a clinical expert system. *JAMIA* 1994;1:428–438.

49. Hershey CO, Porter DK, Breslau D, Cohen DI. Influence of simple computerized feedback on prescription charges in an ambulatory clinic. *Med Care* 1986; 24:472-481.
50. Pozen MW, D'Agostino RB, Selker HP, Sytkowski PA, Hood WB Jr. A predictive instrument to improve coronary-care-unit admission practices in acute ischemic heart disease. *N Engl J Med* 1984;310:1273-1278.
51. Nordic Council on Medicines. Good clinical trial practice: Nordic guidelines. Uppsala, Sweden: Nordic Council on Medicines, December, 1980 (NLN Publication No. 28).
52. Spilker B. Guide to clinical trials. New York: Raven; 1991.
53. Balas EA, Austin SM, Ewigman BG, Brown GD, Mitchell JA. Methods of randomized controlled clinical trials in health services research. *Med Care* 1995;33: 687-699.
54. Cochrane AL. 1931-1971, a critical review with particular reference to the medical profession. In: Teeling-Smith G, Wells N. *Medicine for the year 2000*. London: Office of Health Economics; 1979.
55. Chalmers I, Hetherington J, Newdick M, et al. The Oxford Database of Perinatal Trials: developing a register of published reports of controlled trials. *Control Clin Trials* 1986;7:306-324.
56. Dickersin K. Why register clinical trials? Revisited. *Control Clin Trials* 1992; 13:170-177.
57. Davis DA, Thomson MA, Oxman AD, Haynes RD. Evidence for the effectiveness of CME: a review of 50 randomized control trials. *JAMA* 1992;268:1111-1117.
58. Antman EM, Lau J, Kupelnick B, Mosteller F, Chalmers TC. A comparison of results of meta-analysis of randomized control trials and recommendations of clinical experts. *Treatments for myocardial infarction*. *JAMA* 1992;268:240-248.
59. Light RJ, Smith PV. Accumulating evidence: procedures for resolving contradictions among different research studies. *Harv Educ Rev* 1971;41:420-471.
60. Balas EA, Austin SM, Mitchell JA, Ewigman BG, Kopp KD, Brown GD. The clinical value of computerized information services: a review of 98 randomized clinical trials. *Arch Fam Med* 1996;5:271-278.
61. Begg EJ, Atkinson HC, Jeffery GM, Taylor NW. Individualized aminoglycoside dosage based on pharmacokinetic analysis is superior to dosage based on physician intuition at achieving target plasma drug concentrations. *Br J Clin Pharmacol* 1989;28:137-141.
62. Chiarelli F, Tumini S, Morgese G, Albisser AM. Controlled study in diabetic children comparing insulin-dosage adjustment by manual and computer algorithms. *Diabetes Care* 1990;13:1080-1084.
63. Privitera MD, Homan RW, Ludden TM, Peck CC, Vasko MR. Clinical utility of a Bayesian dosing program for phenytoin. *Ther Drug Monit* 1989;11:285-294.
64. Reid JA, Kenny GNC. Evaluation of closed-loop control of arterial pressure after cardiopulmonary bypass. *Br J Anaesth* 1987;59:247-255.
65. Rodman JH, Jelliffe RW, Kolb E, et al. Clinical studies with computer-assisted initial lidocaine therapy. *Arch Intern Med* 1984;144:703-709.
66. Ziegler DK, Hurwitz A, Hassanein RS, Kodanaz HA, Preskorn SH, Mason J. Migraine prophylaxis: a comparison of propranolol and amitriptyline. *Arch Neurol* 1987;44:486-489.

67. Chambers CV, Balaban DJ, Carlson BL, Ungemack JA, Grasberger DM. Micro-computer-generated reminders: improving the compliance of primary care physicians with mammography screening guidelines. *J Fam Pract* 1989;3:273–280.
68. Fordham D, McPhee SJ, Bird JA, et al. The cancer prevention reminder system. *MD Comput* 1990;7:289–295.
69. Chambers CV, Balaban DJ, Carlson BL. The effect of microcomputer generated reminders on influenza vaccination rates in a university based family practice center. *J Am Board Fam Pract* 1991;4:19–26.
70. McDonald CJ, Hui SL, Smith DM, et al. Reminders to physicians from an introspective computer medical record: a two-year randomized trial. *Ann Intern Med* 1984;100:130–138.
71. Ornstein SM, Garr DR, Jenkins RG, Rust PF, Arnon A. Computer-generated physician and patient reminders: tools to improve population adherence to selected preventive services. *J Fam Pract* 1991;32:82–90.
72. Soljak MA, Handford S. Early results from the Northland immunization register. *N Z Med J* 1987;100:244–246.
73. Austin SM, Balas EA, Mitchell JA, Ewigman BG. Effect of physician reminders on preventive care: meta-analysis of randomized clinical trials. *Proc Annu Symp Comput Appl Med Care* 1994:121–124.
74. Balas EA, Weingarten S, Garb CT, Blumenthal D, Boren SA, Brown GD. Improving preventive care by prompting physicians. *Arch Intern Med* 2000;160:301–308.
75. Krishna S, Balas EA, Spencer, DC, Griffin JZ, Boren SA. Clinical trials of interactive computerized patient education: implications for family practice. *J Fam Pract* 1997;45:25–33.
76. Boren SA, Balas EA, Mustafa Z. Evidence-based patient education for chronic care. *Proc Acad Health Annual Res Meeting, Nashville, TN, June 27–29, 2003.*

Section 2

Applications of Clinical Decision Support Systems in Clinical Practice

8

Clinical Decision Support at Intermountain Healthcare

PETER J. HAUG, REED M. GARDNER, R. SCOTT EVANS,
BEATRIZ H. ROCHA, and ROBERTO A. ROCHA

Decision support technologies are becoming increasingly available to medical practitioners. A variety of programs designed to assist with drug dosing, health maintenance, diagnosis, and other clinically relevant healthcare decisions have been developed for the medical workplace. Increasing ease of access to personal computers is partially responsible for this growth. More important, however, is the growing dependency on computers to maintain part or all of the medical record. This has led to a growing interest in and, in some cases, dependency on, automated medical decision making to support the delivery of economical, quality care.

The Electronic Health Record (EHR) is the primary driver for the growing use of computerized decision tools. The growth in use and sophistication of the EHR has provided a backdrop against which clinical decision support systems (CDSS) appear as a logical consequence.

The EHR itself may be seen as a response to the increasing complexity and volume of both the clinical data associated with an individual patient and the medical knowledge necessary to assimilate and respond to this data. Recent evidence emphasizes the cost of failures to properly integrate the patient's findings with the fruits of medical science. In 1999, the Institute of Medicine estimated that between 44,000 and 98,000 Americans die each year because of medical errors.¹ Computer-based systems have been proposed as a remedy for a large subset of these errors.²⁻⁵

CDSS are often described as a cure for these and other failings in traditional care delivery. Much of the literature that has sparked this awareness comes from research done on an older generation of medical information systems. These systems reside on large mainframe computing hardware. Many of them have been designed to serve hospitals and have supported the patient care given there.⁶⁻⁷ The applications and algorithms that were piloted in these systems have provided the background for the modern decision support technologies, which we see developing and evolving in client/server environments, on personal computers, and on systems based in Internet technologies.

Contributors to the science of applying computer systems to clinical practice include the several sites where hospital-based, medical decision support has been implemented and studied. Among the leaders in these efforts have been groups at the Regenstrief Institute in Indianapolis,⁸ Columbia-Presbyterian Medical Center in New York,⁹ Beth Israel Hospital in Boston,¹⁰ and the HELP System at the LDS Hospital in Salt Lake City.¹¹ Successful efforts to incorporate decision support into order entry systems at the Brigham and Women's Hospital in Boston¹² and Vanderbilt University Medical Center in Nashville¹³ are helping to define the direction that healthcare computing will follow in the future. In this chapter, we will review the experience gained in 25 years of CDSS delivered through the HELP System.

While a great deal can be learned from these hospital-based information systems, a new generation of medical computing environments is evolving. The creators of these environments are not satisfied to provide service for hospitalized patients alone. Instead, their intended scope is the entire healthcare record, covering patients served in both the inpatient and outpatient setting. The systems produced strive to provide a truly longitudinal and comprehensive medical record.

As new systems develop, the infrastructure necessary to provide CDSS is being newly engineered. This provides an opportunity to review the lessons learned in the older systems mentioned above, and to give those lessons form by incorporating them in new healthcare computing implementations. Below, we describe the architecture that we have chosen to incorporate into our newest CDSS as well as the effects of a growing focus on the delivery of new types of medical knowledge.

In this chapter, we focus on the experience of Intermountain Healthcare (IHC), a provider of integrated medical services in the Intermountain West, as an example of two phenomena readily recognized in a variety of healthcare organizations, as they adopt or extend systems designed to replace the paper-based medical record with an electronic one. These phenomena are the continued value of decision support applications in the hospital setting and the growing effort to project and expand the use of these technologies across the entire gamut of clinical care, supporting both new and old CDSS agendas in the inpatient and outpatient setting.

To illustrate decision support in the inpatient setting, we will describe a set of classic applications evolved in the HELP Hospital Information System (HIS) located at the LDS Hospital in Salt Lake City. Teams from IHC, the Department of Medical Informatics of the University of Utah, and commercial partners developed these applications. As a part of our description of decision support, we will discuss the data used and the mechanism through which suggested decisions are communicated to the user. Most CDSS in hospitals depend on simple algorithms to inform and remind users of important clinical data or of medical facts, which may change the decisions they have made or will make. Examples of these include decision

support tools that critique medication orders, and the system for identifying life-threatening laboratory results that are described below.

Below, we also discuss the adaptation of classical CDSS architecture to serve within an enterprise EHR. Rather than focusing on examples, we will endeavor, in this section, to describe the constituents of an environment appropriate for the creation of robust, enterprise CDSS.

An enterprise CDSS implies an enterprise model for knowledge management. This is particularly relevant in light of several new types of decision support being integrated into clinical computing environments.

A key example of these new decision support models is the CDSS associated with Computer-based Physician Order Entry (CPOE). CPOE differs dramatically from the classical decision support environments. These were generally constructed around a vision of the physician's workflow that differed little from the behaviors supported by a wholly paper medical record. CPOE requires an approach to design and delivery that reflects a careful remodeling of the way in which physicians manage a key part of their medical responsibilities, the overall direction of patient care.

The Help System

The overall setting for the CDSS examples described here is the HELP Hospital Information System (HIS). This system is a culmination of more than 25 years of development and testing.¹¹ It currently operates on high availability hardware supplied by the HP NonStop Enterprise Division. Software components of the HELP system have also been installed in many of the 20 hospitals operated by Intermountain Healthcare (IHC). At the LDS Hospital, IHC's central, tertiary care facility, the information system communicates with users and developers through approximately 2,000 terminals and more than 200 printers. The system is interfaced with a variety of other computer systems, including a billing system, a laboratory system, a medical records system, a digital radiology system, and a collection of local area networks (LANs) used by a variety of departments for local research and departmental management functions.

The HELP System consists of an integrated clinical database, a frame-based medical decision support system, programs to support hospital and departmental clinical and administrative functions, and the software tools needed to maintain and expand these components. The integrated clinical database contains a variety of patient data (Table 8.1) kept online during the patient's stay to allow review by health-care professionals at terminals throughout the hospital. These terminals allow the entry of pertinent clinical data into the HELP system by personnel who are involved in patient care. In addition, automated systems capture clinical information directly from monitors and other instruments in the hospitals' ICUs.

TABLE 8.1. Clinical data routinely captured by the HELP hospital information system (partial list).

Chemistry	Hematology
Medications	X-ray Findings
Allergies	Dietary Information
Blood Gases	Surgical Procedures
Electrocardiograms	ICU Monitoring
Intake/Output	Pulmonary Function
Demographic Information	Microbiology
Cardiac Catheterization Data	Respiratory Therapy Notes
Biopsy Results	Nursing Data
Select Physical Examination	Pathology Department Data
Admit/Discharge Information	History and Physical Exam Reports
Consult Reports	Procedure Reports

Use of the HELP system as a medical expert system has been a major focus of research since the system's inception. The result has been a set of embedded expert system development tools. The HELP System contains a decision support subsystem based on a modular representation of medical decision logic in frames.¹⁴ These modules are used to: (1) define the data used in making the target medical decision; and (2) encode the logic that converts the raw data into the proposed decision. Decisions encoded in these modules resemble small computer programs written in a Pascal-like language. They are each designed to represent a single simple decision capable of activation in a number of ways. The language supports either simple or multiple outputs from a frame. This flexibility can be used to create more complex modules capable of deriving several distinct decisions from the same data.

This set of tools has led to the successful development of expert systems in blood gas interpretation,¹⁵ intensive care settings,¹⁶ and medication monitoring,¹⁷ to name a few. The HELP System hardware and software environment has provided the setting for the implementation and testing of most of the decision support examples described below.

The history of decision support in the HELP System extends more than 25 years into the past. This classic hospital information system includes two types of CDSS systems. The first type focuses on narrowly circumscribed medical conditions. The logic is typically simple and the data requirements modest. The Critical Laboratory Alerting System described below is an example of this type.

The second type of CDSS is much less common. This type of tool attempts to discriminate among a group of important *diagnostic* entities using raw medical data. Diagnostic systems often attempt the challenging task of managing large degrees of uncertainty using pattern matching algorithms. Several of these systems have been, or are being, tested in the HELP

environment. Below, we describe experience with three of the experimental diagnostic applications.

Categories of Decision Support Technologies

Independent of the environment in which they are used, two elements of medical decision support applications are critical to their success. These are: (1) the mechanism by which the systems acquire the data used in their decision algorithms; and (2) the interface through which they interact with clinicians to report their results. These considerations have led us to describe different categorizations of decision support.¹⁸ Although somewhat arbitrary, this categorization captures the idea that different models of computerized assistance may be needed for different types of clinical problems.

The four categories are:

1. Processes which respond to clinical data by issuing an alert;
2. Programs activated in response to recorded decisions to alter care (typically new orders); these applications work by critiquing the decision and proposing alternative suggestions as appropriate;
3. Applications that respond to a request by the decision maker by suggesting a set of diagnostic or therapeutic maneuvers fitted to the patient's needs;
4. Retrospective quality assurance applications where clinical data are abstracted from patient records and summary decisions about the quality of care are made and fed back to caregivers.

We will describe the first three types in this chapter.

Alerting Systems

Alerting processes are programs that function continuously, monitoring select clinical data as it is stored in the patient's electronic record. They are designed to test specific types of data against predefined criteria. If the data meet the criteria, these systems alert medical personnel. The timing and character of the messages vary with the alerting goals.

A typical example is a subsystem implemented within the HELP System that monitors common laboratory results and detects and alerts for potentially life-threatening abnormalities in the data acquired. This type of application is notable for the simplicity of its decision logic as well as for the magnitude of its potential impact.

The HELP System captures results from the clinical laboratory through an interface to a dedicated laboratory information system (LIS). The results are collected and returned to the HELP System for storage in the clinical record as soon as they are collected and validated in the LIS.

Laboratory results are reviewed by personnel engaged in patient care both through terminals connected to the HELP System and through a variety of special and general-purpose printouts, such as rounds reports generated by the HELP System. The “times” when the data are reviewed have only a loose relationship to the “times” when these data become available. Instead, the principal determinant of the review time is typically the work schedules of the physicians and nurses involved with the patient. The physician, for instance, may visit the hospital twice a day for rounds and review patient data only during those times unless some aspect of the patient’s condition prompts a more aggressive approach.

Under these circumstances, abnormalities in laboratory results, especially those that are unexpected, may not receive the timely attention they deserve. In particular, unexpected laboratory abnormalities may go unseen for hours until a nurse or physician reviews them during their routine activities. Or, as some authors have noted, they may be missed entirely.^{19,20}

As a response to this disparity, Karen Bradshaw-Tate and her associates have described an experiment with a Computerized Laboratory Alerting System (CLAS) designed to bring potentially life-threatening conditions to the attention of caregivers.²¹⁻²⁴ This system was constructed by reducing a set of 60 alerts developed during a previous pilot system²⁵ to the 10 most important (Table 8.2).

Six medical experts from the disciplines of surgery, cardiology, internal medicine, and critical care participated in the development of these alerts and the system used to deliver them. The alerts chosen were translated into computer logic and tested to determine that the logic functioned properly. Data from previously admitted patients were used to refine and test the logic.

Once the logic was deemed acceptable, an experiment was designed to evaluate the effect of the system on several intermediate outcome

TABLE 8.2. Alerts for which computerized alerting logic was created.

Alerting Condition	Criteria
Hyponatremia (NAL)	Na ⁺ < 120 mEq/l
Falling Sodium (NAF)	Na ⁺ fallen 15+ mEq/l in 24h and Na ⁺ < 130mEq/l
Hypernatremia (NAH)	Na ⁺ > 155 mEq/l
Hypokalemia (KL)	K ⁺ < 2.7 mEq/l
Falling Potassium (KLF)	K ⁺ fallen 1+ mEq/l in 24h and K ⁺ < 3.2mEq/l
Hypokalemia, patient on digoxin (KLD)	K ⁺ < 3.3mEq/l and patient on digoxin
Hyperkalemia (KH)	K ⁺ > 6.0mEq/l
Metabolic Acidosis (CO ₂ L)	CO ₂ < 15 and BUN < 50 or CO ₂ < 18 and BUN < 50 or CO ₂ < 18 (BUN unknown) or CO ₂ fallen 10+ in 24hr. and CO ₂ < 25
Hypoglycemia (GL)	Glucose < 45 mg%
Hyperglycemia (GH)	Glucose > 500 mg%

measures. Two approaches were tested for delivering the alerts. The first of these techniques was tested on a single nursing division to determine its acceptability. A flashing yellow light was installed in the division, and whenever an alert was generated for a patient in that division, the light was activated. It continued to flash until the alert was reviewed and acknowledged on a computer terminal.

The second approach was less intrusive to the nursing staff. Whenever anyone accessed the program used to review a patient's laboratory results, any unacknowledged alerts for that patient were immediately displayed along with the data that had triggered them.

The results of this type of intervention were tested in three ways. First, appropriateness of treatment was evaluated. The alerting system was shown to result in a significant increase in appropriate therapy for conditions involving abnormalities of Na^+ , K^+ , and glucose. Second, time spent in the life-threatening condition with and without the alerting system was examined. Finally, the hospital length of stay was examined. A significant improvement in this parameter was also noted for the patients with abnormalities of Na^+ , K^+ , or glucose.

This type of decision support intervention is becoming increasingly common as hospital information systems evolve.²⁶ In the inpatient environment where the severity of illness is steadily increasing, the possibility of better alerting has the potential to improve quality of patient care.

Interestingly, the system for alerting on critical laboratory values has been re-implemented in recent years. The IHC laboratory that processes the inpatient laboratory values also serves a variety of locations into which the HELP System does not reach, notably a large number of outpatient clinics. Based upon the value of this type of intervention, Laboratory Services has instituted the process of having personnel telephone ordering physicians or other caregivers whenever critical laboratory values are detected. Thus, the limitations of a model that was restricted to select inpatient locations have been circumvented.

The developing enterprise information system, parts of which are described below, will provide yet another way to avoid the limitations of an inpatient system. This system can reach the caregivers associated with outpatients as well as inpatients, and it invites a re-implementation of the computerized version of this system in a way that provides comprehensive coverage. The evolving capability to move alerting to an outpatient setting is illustrated by the example that follows.

A recent alerting application designed to work in the outpatient setting is among the first to take advantage of a new, enterprise CDSS infrastructure. This application automates a part of the Chronic Anticoagulation Clinic's (CAC) anticoagulation protocol. This clinic manages patients that are taking anticoagulation drugs (principally Coumadin) for extended periods of time. The objective is to maintain each patient's International Normalized Ratio (INR) within a range specified for the patient. A key component is a

rule-based system that monitors coagulation studies for compliance with these goals and presents alerts to the clinical user through a computerized in-box. Alerts for dangerously altered INRs are also sent to the clinic nurse practitioner's pager so that immediate action can be taken.

The CAC protocol has been working since June 2003, and since then the clinic has come to rely completely on the alerts generated by the protocol. They replace a paper-based process and couple the prescribing practice of the physicians (captured in the enterprise EHR) with the clotting test results captured in the clinical laboratory that reflect the effectiveness of this therapeutic intervention.

Critiquing Systems

In the alerting example described above, the computer system responded to abnormalities in the data as they entered into the database by prompting those caring for the patient to intervene. In contrast, critiquing processes begin functioning when an order for a medical intervention is entered into the information system. Such methods typically respond by evaluating an order and either pointing out disparities between the order and an internal definition of proper care or by proposing an alternative therapeutic approach. Below, we describe a critiquing subsystem that specifically targets orders for blood products.

Over the years, it has become apparent that the transfusion of blood products is an important, often life-saving, therapy and that these same blood products must be ordered and administered with care. Not only are there significant reasons for anxiety concerning diseases that can be transmitted during transfusions, but also the limited supply and short shelf life of blood products make them a scarce resource to be used sparingly. In 1987, the Joint Commission for the Accreditation of Healthcare Organizations (JCAHO) began to require healthcare institutions to develop criteria for the use of blood products and to carefully monitor compliance with these criteria.

At the LDS Hospital, the response to these requirements was to develop a computer system designed specifically to manage the ordering of transfusions and to assist in ensuring compliance with criteria for proper use of blood products.²⁷⁻³⁰ A central premise of the system design was that all orders would be entered into the computer and that physicians or nurses would enter all blood orders.

Embedded in the blood-ordering program is a critiquing tool designed to ascertain the reason for every transfusion and to compare the reason against strict criteria. The approach used provides information specific to the type of transfusion planned. For instance, when an order is made for packed red blood cells, the criteria in Table 8.3 are used to critique the order.

TABLE 8.3. Simplified criteria for ordering red blood cells.

Hemoglobin < 12 g/dl or hematocrit < 35% if age \geq 35 years
Hemoglobin < 10 g/dl or hematocrit < 30% if age < 35 years
Oxygen saturation (SaO ₂) < 95%
Active bleeding
Blood loss > 500 ml
Systolic blood pressure < 100 mmHg or heart rate > 100 bpm
Adult respiratory distress syndrome (ARDS)

The process of entering an order into this system includes several points at which information bearing on the propriety of giving blood products is displayed. As a first step, the physician is shown the blood products ordered in the last 24 hours. This is followed by a display of the applicable laboratory data. Then the user chooses the specific blood products required along with the number of units and the priority (stat, routine, etc.). At this point, the user is asked to document the reason for the order. A list of reasons, specific to the blood product chosen, is displayed, and the user chooses the appropriate rationale for the intervention. The computer then applies the stored criteria and determines whether the order meets the hospital's guidelines.

If the guidelines are met, the order is logged and the blood bank and nursing division are informed electronically and via computer printout. If the criteria are not met, the user is presented with a message stating the applicable criteria and relevant patient data. The physician or nurse may optionally decide to place or cancel the order. If the order is made, he or she is required to enter the reasons for the decision to override the system.

The criteria used are the result of a consensus effort by the LDS Hospital medical staff. The criteria were developed using primarily published guidelines but with some adaptations for local conditions (altitude of 4,500 feet). The criteria have undergone several modifications based on experience as well as new definitions of standards for these therapies.

One way of measuring the effectiveness of the system's various critiquing messages is to examine the frequency with which the process of ordering blood products is terminated as a result of the feedback. During one six-month period, the ordering program was entered and then exited without an order 677 times. This was 12.9% of the total uses. We estimate that one-half of these exits represent decisions not to order blood products based on feedback from the program.

The program relies heavily on the integrated clinical database in the HELP System. It accesses data from: (1) the admitting department; (2) the clinical laboratory; (3) surgical scheduling; (4) the blood bank; and (5) the orders entered by nurses and physicians.

The blood-ordering program described above contains processes that support computerized critiquing. The program responds to interventions chosen by the physician by analyzing the order and, if appropriate, suggesting reasons to alter the therapeutic plan.

The process used by the blood-ordering program is different than that used in the alerting application in that it involves a dialogue with the user. As a result, the critique can provide a series of informational responses designed to assure that the user is fully aware of the status of the patient as well as of accepted guidelines governing blood product usage.

Historically, physician use of generalized computerized order entry programs has been limited. However, modern order entry programs are being designed to encourage their use by physicians. A part of this encouragement is based on the ability of these programs to critique orders. Physicians often appreciate the ability of an automated ordering system to give feedback on proper dosing and accepted care protocols as they make their interventional decisions. Opportunities for a constructive interaction between the computer and the clinician are clearly growing, and applications that critique medical decisions can contribute to this growth.

Suggestion Systems

The third category of computer applications designed to support medical decision making is potentially the most interactive. This group of processes is designed to react to requests (either direct or implied) for assistance. These processes respond by making concrete suggestions concerning which actions should be taken next.

Unlike alerts, action oriented messages from these systems are expected. Clinicians would typically call up a computer screen, enter requested data, and wait for suggestions from these systems before instituting a new therapy. Unlike critiquing systems, the physician need not commit to an order before the program applies its stored medical logic. Instead, the program conducts an interactive session with the user during which a suggestion concerning a specific therapeutic decision is sought. The system then reviews relevant data, including data that has been requested from the user, and formulates a suggestion for an intervention based on the medical knowledge stored in its knowledge base.

The example below is, in many ways, typical of suggestion systems. It functions in the realm of ventilator therapy and has been implemented in increasingly more sophisticated forms in intensive care settings at the LDS Hospital since 1987.

As a tertiary care setting, LDS Hospital sees a large number of patients with respiratory failure. One of the more difficult of these problems is that of Adult Respiratory Distress Syndrome (ARDS). This disease can complicate a number of other conditions, including trauma, infectious disease,

and shock. The usual therapy includes respiratory support while the underlying pulmonary injury heals. Unfortunately, overall mortality for ARDS had remained at about 50% for many years. For the subset of ARDS patients who manifest severe hypoxemia, the mortality had been approximately 90%.

The study of computer protocols for delivering care to ARDS patients was a side effect of research into the effectiveness of a new therapeutic intervention of this difficult disease. In the early 1980s, research began to suggest that external membrane devices that bypassed the lungs to remove carbon dioxide (CO₂) directly from a patient's body might improve survival in the most severely ill ARDS patients. Physicians at the LDS Hospital wanted to study this new approach in a rigorously controlled clinical trial. They chose to do an experiment with a test group that received the external lung treatment and a control group that did not receive the treatment. However, the researchers were aware that the management of ARDS differed from patient to patient, depending on the course the disease followed, and the training and previous experience of the physicians and staff caring for the patient. For this reason, they decided to standardize care by strict adherence to predetermined treatment protocols.

At first, they developed a set of paper protocols. As the protocols became more complex, it became clear that they would be difficult to follow manually. Therefore, it was decided to computerize them. The result was a set of computerized rules that were designed to direct, in detail, the management of patients in both the test and control branches of a study of extracorporeal CO₂ removal (ECCO₂R).³¹⁻³³ While the rules were designed initially for this research, they were soon made general enough that they could be used in the management of other patients requiring ventilator support.

The protocols were created by a group of physicians, nurses, respiratory therapists, and specialists in medical informatics. The initial study period was to be 18 months. Subsequent development concentrated on first eliminating errors in protocol logic, second on extending the scope of these tools, and finally on reworking behavioral patterns in the intensive care setting so that the protocols could be effectively implemented.

The protocol system devised was used successfully during the ECCO₂R study. The study was terminated after 40 patients were treated, 21 with ECCO₂R and 19 with conventional therapy. At that time, there were eight survivors in the conventional therapy group (42%) and seven in the ECCO₂R group (33%).³³ The study group concluded that there was no significant difference between ECCO₂R and conventional treatment of severe ARDS. However, the 42% survival in the control group was unexpected. Reported survivals in these severely ill patients were less than 15%. The results led the researchers to suspect that the quality and uniformity of care provided through the use of computerized protocols had resulted in an important improvement in patient outcomes.

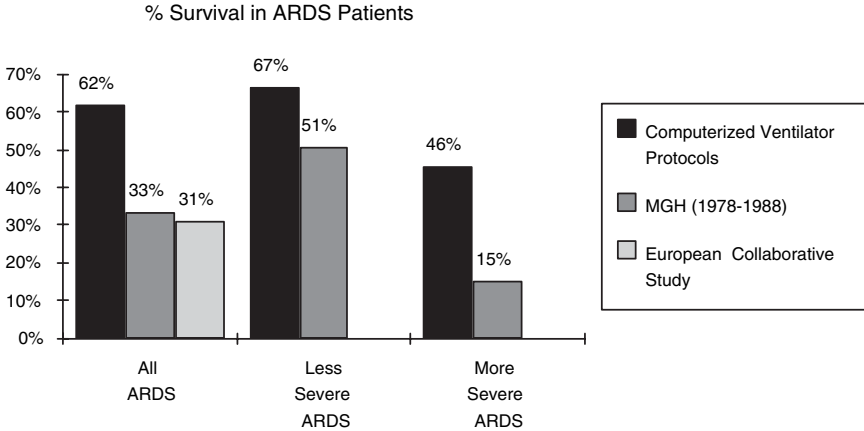


FIGURE 8.1. Comparative results for groups managing ARDS patients.

As a consequence, development and study of these protocols has continued. Figure 8.1 summarizes the results of their use in 111 LDS Hospital patients, and compares these results to those of two other groups Massachusetts General Hospital (MGH) and a group in Europe (the European Collaborative Study) interested in the problem of treating ARDS. It is becoming increasingly clear that the standardization of complex ventilator care decisions possible with computers has a pronounced benefit for patients.

It should be noted that here we have focused the definition of systems for suggesting therapeutic interventions quite narrowly. We have limited our example to a system that responds with a suggestion when the clinician has explicitly or implicitly requested one. Such a computerized decision support process is an area in which we are continuing to explore better ways to interact with clinicians and better ways to capture and encode protocol knowledge.

Diagnostic Decision Support in the Help System

The examples above have stressed different approaches to the activation of medical decision support logic and to the delivery of the resulting decisions to the computer user. Below, we change our focus. One of the greatest challenges for a computerized medical decision system is to participate usefully in the diagnostic process. Diagnostic decision support systems (DDSS) differ from the CDSS described above. Typical decision support systems can draw attention to specific data elements and/or derive therapeutic suggestions from these elements. Such applications offer assistance in the basic

recognition processes and can categorize patients by pathophysiologic condition. On the other hand, the diagnostic process is a preliminary step to suggesting therapeutic interventions. Computerized diagnostic decisions are generally involved with different goals, interfaces, and decision algorithms than the applications previously described.

Two types of diagnostic applications are described. They differ in the degree with which the developers have solved the problem of providing a clinically useful service. The first type represents a group of applications that, using a set of raw clinical data, attempt to standardize various diagnostic categorizations that impact discrete therapeutic decisions. Three HELP System examples are discussed.

The second group of diagnostic processes described comes from the family of applications that attempt to simulate the more extensive and flexible diagnostic behavior of physicians. Those discussed here represent preliminary research whose clinical applicability remains to be determined. The status of these applications in terms of preliminary data and experience limited to a research and development environment are described.

Proven Diagnostic Applications

A number of applications residing in the HELP system can, through the use of various diagnostic strategies, affect patient care. Below we describe three of these applications. The first is an application that evaluates patient data to detect adverse drug events. The second is a tool that recognizes nosocomial infections. The third is a computerized assistant that informs and advises physicians as they undertake the complex task of determining how to treat a patient with a possible infection.

Adverse Drug Events

Adverse drug events (ADEs) are defined by the World Health Organization as “any response to a drug which is noxious, unintended, and which occurs at doses normally used in man for the prophylaxis, diagnosis, or therapy of disease.” ADEs can range in severity from drowsiness or nausea to anaphylaxis and death. It has been estimated that in the United States that drug-related morbidity and mortality costs more than \$136 billion per year.³⁴

The process of recognizing ADEs differs from that of drug monitoring at the time of drug dispensing; this latter process has become a standard part of computerized pharmacy systems. The alerting systems embedded in modern-day pharmacy dispensing systems typically evaluate ordered medications against a list of contraindications based on known allergies, expected reactions with other patient medications, or the information from

the clinical laboratory that can be expected to affect the drugs given or the dosage of those medications. In contrast, the goal of an ADE detection system is to determine the existence of a drug reaction from the patient data collected during the routine documentation of patient care.

An ADE recognition subsystem has been implemented in the HELP system.³⁵⁻³⁶ This ADE subsystem continuously monitors patients for the occurrence of an ADE. The system does so by inspecting the patient data entered at the bedside for signs of rash, changes in respiratory rate, heart rate, hearing, mental status, seizure, anaphylaxis, diarrhea, and fever. In addition, data from the clinical lab, the pharmacy, and the medication charting applications are analyzed to determine possible ADEs.

The system evaluates all of the patients in the hospital and generates a daily computer report indicating which patients have possible ADEs. A clinical pharmacist then follows up on these patients and completes the evaluation using a verification program. This program provides a consistent method of completing the diagnostic process. A scoring system (the Naranjo method) is used to score the ADEs as definite (score ≥ 9), probable (score 5-8), possible (score 1-4), or unlikely (score 0).³⁷ The physicians caring for each patient are notified of confirmed ADEs by the pharmacist who does the evaluation.

The existence of an application for diagnosis of ADEs has increased the frequency with which these events are recognized and documented in the hospital setting. Using a voluntary reporting method, nine ADEs were recorded in the one-year period from May 1, 1988 to May 1, 1989. In the period from May 1, 1989 to May 1, 1990, while the program was in use, 401 adverse drug events were identified.

An additional effect of this program appears to be a reduction in the number of severe ADEs seen. During the year beginning in January of 1990, 41 ADEs occurred. In this time frame, physicians were notified of verified ADEs only if they were classified as severe or life threatening. In two subsequent periods (the year of 1991 and the year of 1992) early notification of physicians was practiced for all severities of ADE. Numbers of severe ADEs decreased to 12 and 15 during the follow-up time periods ($p < 0.001$).

In an effort to understand the impact of the drug reactions that were the target of this application, the costs of ADEs were examined. In studies that used the computer tools described above, investigators found that length of hospital stay for patients with ADEs was increased by 1.91 days and that costs resulting from the increased stay were \$2,262. The increased risk of death among patients experiencing ADEs was 1.88 times.³⁸ Thus, the cost savings and impact on quality of care in reducing ADEs was substantial.

These tools leverage the fact that the majority of the data necessary for their function is available in the HELP system's integrated database. They illustrate the potential for computerized diagnostic applications to impact

patient care not just by assisting with the choice of interventions, but also by focusing clinical attention on those cases where the interventions chosen have put the patient at risk.

Nosocomial Infections

In the previous example, a rule-based system was used to suggest the diagnosis of adverse drug events for a group of patients undergoing therapy in the hospital. Another application in use at the LDS Hospital is designed to recognize nosocomial, or hospital acquired infections.³⁹ The program serves a need recognized by the JCAHO that requires ongoing surveillance for hospital-acquired infections.

The process of detecting nosocomial hospital infections serves a recognized clinical purpose. Control measures based on this information are believed to be important in interrupting the spread of hospital-acquired infections. Evidence suggests that intensive surveillance programs may be linked to reduced rates of infection. However, the process can be expensive. Traditional techniques require infection control personnel to manually screen all appropriate patients on a routine basis.

The computerized surveillance system used in LDS Hospital relies on data from a variety of sources to diagnose nosocomial infections. Information from the microbiology laboratory, nurse charting, the chemistry laboratory, the admitting office, surgery, pharmacy, radiology, and respiratory therapy are used. Once each day, a report is produced detailing the computer's findings. This report can be used to follow up on the patients for whom there is evidence of nosocomial infection.

In studies done to compare the computer-based surveillance of nosocomial infections to the traditional, manual approach, 217 patients were determined to be possible victims of hospital-acquired infection (out of 4,679 patients discharged in a two-month period). This included 182 patients identified by the computer and an overlapping 145 patients recognized by traditional means. Of these patients, 155 were confirmed to have nosocomial infections.

For the group of 155 patients, the computer's sensitivity was 90% with a false positive rate of 23%, while the infection control practitioners demonstrated a sensitivity of 76% and a false positive rate of 19%. When the hours required to use each approach were estimated, the computer-based approach was more than twice as efficient as the entirely manual technique.

The nosocomial infection tool, like the ADE recognition system, uses Boolean logic in a relatively simple diagnostic process. In an effort to extend the process of managing hospital-acquired infections, an extension to the infection control system was developed. The goal of the enhancement was to predict which patients were likely to contract a nosocomial infection in the hospital in the future. The tool is based on different decision algorithms. Data from patients with infections acquired in the hospital were combined

with data from a control set of patients, and a group of statistical programs were used to identify risk factors. Logistic regression using these risk factors was used in the development of tools that could estimate the risk of hospital-acquired infection for inpatients. The resulting system is capable of predicting these infections in 63% of the population who are ultimately affected.⁴⁰

Antibiotic Assistant

The third application in this group is an example of a multipronged approach to the task of supporting medical decision making. As a part of ongoing research into the use of computers in medical care, the Infectious Disease Department at LDS Hospital developed a tool to help clinicians make informed decisions concerning the administration of antibiotics.^{41,42} The “antibiotic assistant” application provides three basic services. First, it assembles relevant data for the physicians so they can determine whether a specific patient is infected and what sorts of interventions might be appropriate. Information such as the most recent temperature, renal function, and allergies are presented. Second, the system suggests a course of therapy appropriate to that patient’s condition. Finally, the program allows the clinician to review hospital experience with infections for the past six months and the past five years. One of the options of the program allows the clinician to review the logic behind the computer’s suggestions while another presents brief monographs on the appropriate use of each antibiotic in the hospital formulary.

The diagnostic processes embedded in this application are derived from data extracted from the HELP system and analyzed on a monthly basis. The goal of the analysis is to define the probability of each potential pathogen as a causative agent for a certain class of patient. Six clinical variables are used in this process. These variables were identified through a statistical analysis of 23 proposed data elements. They include the site of infection, the patient’s status (inpatient or outpatient), the mode of transmission (community- or hospital-acquired), the patient’s hospital service, the patient’s age, and the patient’s sex.

The result of this monthly analysis is an assessment of the likelihood of each pathogen for every combination of the patient-related variables. For example, once the analysis is complete, the percentage of hospital-acquired bacteremias due to *Escherichia coli* in male patients age 50 or less who are on the cardiovascular service will be stored in the program’s knowledge base. The analytic programs also evaluate susceptibility data to determine which antibiotics are likely to cover the most probable pathogens for each combination of patient variables.

This probabilistic knowledge is then filtered through a set of rules created by infectious disease experts. These rules adjust the output of the first phase to include criteria representing basic tenets of antibacterial therapy. For

example, the susceptibility information garnered from the historical data would be updated to indicate that amikacin should be used only for infections due to gram-negative organisms.

The resulting knowledge base is used by the antibiotic assistant program to make presumptive diagnoses of infectious organisms and to suggest treatments appropriate to these organisms. It remains up-to-date through monthly updates of its knowledge base. By offering the monographs and explanations mentioned above and by allowing the clinicians to browse its knowledge base, it provides large amounts of information in addition to its suggestions.

Research into Complex Diagnostic Applications

The systems described above have had a clear and measurable effect on improving health care provided in the hospital setting. The dream of even more sophisticated and inclusive systems were presented more than 30 years ago. In 1959, Ledley and Lusted described the application of methods from the realm of symbolic logic and statistical pattern recognition to problems in medicine.⁴³ They proposed that these tools be used to assist in the diagnostic process and in other problems involving medical decision making. Computer systems were the enabling technology that was predicted to bring these tools to the bedside.

A variety of researchers have accepted the challenge of Ledley and Lusted and produced experimental systems designed to diagnose a variety of illnesses. A number of these systems are mentioned elsewhere in this book. Within the HELP system, researchers have created and tested several DDSS. Two of these are described below.

An important portion of the value of computerized diagnostic tools lies in the development of well-designed models of the diagnostic process to assist in the complex clinical decision-making tasks. Physicians clearly exercise their diagnostic knowledge not only when they assign a diagnostic label to a patient, but also during processes as diverse as reading medical reports and critiquing the clinical behavior of their peers. Below, we give examples of experimental systems that: (1) assist with data collection; and (2) help assess the quality of medical reports.

The applications described below benefit from a long-standing interest in Bayesian techniques for probability revision among researchers using the HELP system. For more than 20 years, the HELP system has contained a frame-based decision support subsystem capable of capturing and employing Bayes' equation to assess probabilistically the support for diagnoses provided by various combinations of clinical data.¹⁴ Approaches to decision support, such as those described in Chapter 2 of this book, have been and continue to be key areas of research in the HELP medical informatics community.

Assisting Data Collection

Efforts to direct data collection in the HELP system have concentrated on the patient history. The goal has been to identify tools that could effectively collect a medical history appropriate for use in diagnostic decision support applications. While earlier efforts focused on history appropriate to a wide variety of diseases,⁴⁴ more recent efforts have focused on acquiring data bearing on pulmonary diseases.^{45,46}

Three techniques for collecting the history were explored. The first was a simple branching questionnaire. This approach takes full advantage of the hierarchical relationship between more and less specific questions. For instance, if the question "Have you had chest pain with this illness?" was answered "Yes," then more specific questions such as "Is your chest pain brought on by exertion?" were asked. Alternately, if the answer to the first question were "No", the more specific questions would not be asked.

The second technique has been called decision-driven data acquisition (DDA). With this technique, a frame-based, Bayesian expert system analyzes all data available at any point in the patient interview. The individual disease frames determine which additional information is needed to evaluate the likelihood of the particular disease. Each frame proposes one or more questions. From this list, a supervisory program selects a group of five questions, which are then presented to the patient. The system passes through this cycle multiple times until criteria are met indicating that no additional data are needed.

A third approach has also been tested. It is similar to the DDA method except that it was adapted for use in a setting where the patient was not present at a computer terminal. The approach begins when a paper questionnaire containing screening questions is presented to a patient. Staff members enter the answers into the computer, and the patient's data are compared to the diagnostic frames. The questions are scored in a filtering process, and then from 0 to 40 additional questions are printed for the patient to answer. After the patient answers these additional questions, the answers are entered into the computer and the process is completed.

The branching questionnaire mode of data collection and the DDA mode were tested on inpatients at the LDS Hospital. Fifty patients took a DDA managed history and 23 received a history managed by the branching questionnaire program. Figure 8.2 illustrates the results.

On average, the DDA mode took a significantly ($p < 0.05$) shorter time to run (8.2 minutes) and asked significantly fewer questions (48.8 questions) than did the branching questionnaire (19.2 minutes and 137 questions, respectively). The two-stage, paper questionnaire was tested separately on patients coming to the X-ray department for chest X-rays. It appeared to perform similarly to the interactive DDA mode. It should be noted that there was no significant difference between the techniques in terms of

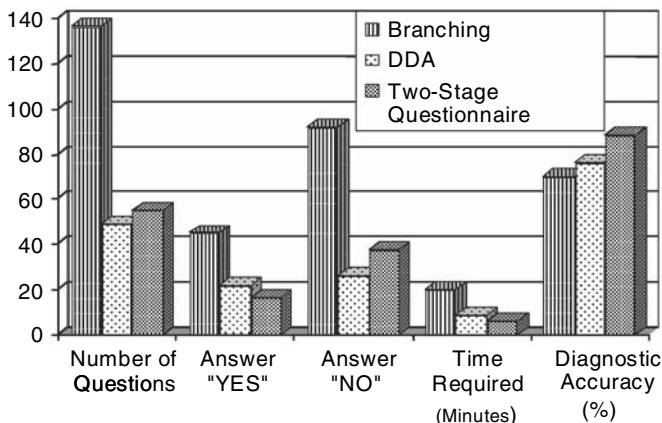


FIGURE 8.2. A comparison of techniques for collecting the patient history.

diagnostic accuracy. Using history alone, all three succeeded in placing the patient's correct disease in a five-member differential diagnostic list from 70–88% of the time.

Assessing the Quality of Medical Reports

A second example of an alternative use of diagnostic knowledge comes from a study of result reporting in the radiology department. The central goal of this project was to develop a technique for measuring the quality of X-ray reporting without requiring the review of radiographs by multiple radiologists. This is in contradistinction to typical approaches for evaluating the accuracy of radiologists. Typically, audit procedures in the radiology department require multiple readings of a select set of X-rays.^{47–51} The results of the repeated readings are used to define a “gold standard” for the films. Then the individual radiologists are compared to the gold standard.

The technique developed as a part of this project was based on a simple premise. Each examination was a test of the radiologist's accuracy. Instead of comparing the abnormalities reported to a standard formulated through multiple readings, the description in the report was evaluated in comparison to the patient's overall diagnostic outcome. In the case of chest X-rays, the standard was the list of final diagnoses (ICD-9 codes) integrated into the patient's record at the time of discharge. The report generated by the radiologist was successful to the extent that it supported the process that led to one of the discharge diagnoses.

While a variety of algorithms can be used to link the findings represented in the X-ray report to the final diagnosis, we have demonstrated the success

of a variation on Shannon Information Content in discriminating among physicians reading chest X-rays. Shannon Information Content⁵² is a mathematical formalism for assessing the informational value of messages. We have modified it to provide a measure of the information produced by the radiologists as *they* interpret an X-ray. The assumption inherent in this usage is that the information contained in an X-ray report can be expected to alter the likelihood of the various diseases that a patient might have. Information Content is calculated from the change in probability of these diseases.

For this technique to work, a diagnostic system was required that was capable of discriminating among diseases producing abnormalities on the chest radiograph. The information content was calculated from the change in disease probability induced by the findings recorded in the chest X-ray report. A Bayesian system provided the required probabilities.

Our evidence for the success of this technique came from two studies. In the first, we used expert systems technologies to demonstrate discrimination in a controlled experiment.⁵³ In this experiment, five X-ray readers read an identical set of 100 films. The assessment produced by the diagnostic logic program gave results consistent with the differing expertise of the readers and similar to the results of a more standard audit procedure.

In a second study of this audit technique, we extended the test environment into the realm where we hope to use it clinically.⁵⁴ We tested a group of radiologists following their standard procedure for interpreting radiographs. Each chest X-ray was reviewed, the report dictated and transcribed only once, as is typical with most radiologists' daily work. The goal of the study was to test the ability of a knowledge-based approach to measure the quality of X-ray reporting, without requiring repeated reading of the radiographs.

This technique used a modified version of the Shannon Information Content measure, and was designed to assess both the positive information contributed by X-ray findings relevant to a patient's disease, and the negative information contributed by findings which do not apply to any of the patient's illnesses. X-ray readers were compared based on the bits of information produced. We used 651 chest X-ray reports, generated by a group of radiologists, that were compared to the patients' discharge diagnoses using a measure of information content. The radiologists were grouped according to whether they had received additional (post residency) training in chest radiology. The "trained" radiologists produced 11% more information than the "untrained" radiologists (0.664 bits as opposed to 0.589 bits, significant at $p < 0.005$).

The average information content calculated successfully discriminated these groups. However, it is an overall measure. Examination of the interaction between the groups of radiologists and disease subgroups indicates that the score can also discriminate at the level of different diseases ($p < 0.05$). This suggests that the technique might not only discriminate overall

quality of X-ray interpretation, but it might also be of use at pinpointing the specific diseases for which an individual radiologist may be failing to generate effective information.

Infrastructure for an Enterprise Clinical Diagnostic Support Systems

In order to build and test the variety of CDSS applications described above, an environment conducive to the development of decision support applications is necessary. The HELP system served this role for more than two decades. During that time, the development and maintenance of an infrastructure, designed to sustain an effective CDSS, became a central tenet of the system.

Now, a new medical computing environment is replacing the HELP System as the core of IHC's Electronic Health Record. This system is known as HELP2. Based on our experience with the HELP system,⁵⁵ a new decision support infrastructure has been developed for this new platform.⁵⁶ This infrastructure is comprised of five main modules: *data-drive*, *time-drive*, *rule node*, *dispatch node*, and *configuration manager*. Figure 8.3 illustrates the design.

Data-drive is the module responsible for activating the rules whenever any clinical data are stored in the database (new, updated, or logically deleted). Whenever data are stored in the clinical data repository, a copy is forwarded to the *data-drive* module. The data instances are filtered using a configuration file that identifies data for which decision rules exist. Only those that match continue to be processed. They are transformed into a standard data representation and sent to the *time-drive* module. This allows

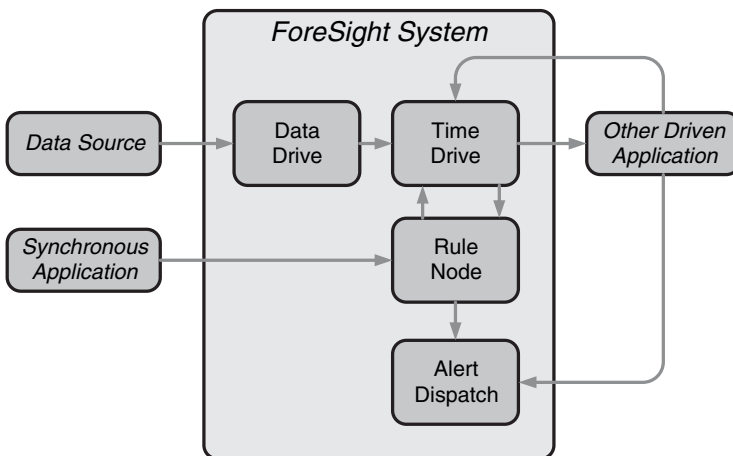


FIGURE 8.3. CDSS infrastructure.

a temporal offset between the receipt of the data and the execution of the rules.

Data that arrive in the *time-drive* module can be held there for a predetermined amount of time before they are delivered to the *rule node*. The objective is to be able to activate the rules at certain times of the day, or after a certain period of time. The holding time can be from seconds to years. In most cases, data that come from the *data-drive* typically have no waiting time, and are immediately delivered to the *rule node*.

The *rule node* was designed to allow wide choice in the methods used for processing the data. It can run different inference engines, allowing different representations of knowledge. We have tested with rule in pure java code as well as logic-executed in third-party inference engines. The *rule node* receives the data and verifies which rules or protocols should be executed. Besides *data-drive* and *time-drive*, the rule node can also be activated synchronously, i.e., directly by an application. If the activating application were an interactive application, it would be able to activate a needed rule set directly and receive in reply the computed decisions. These could then be presented to the waiting user.

If additional data are necessary to execute triggered rules, the data are retrieved from the database and converted into the same common data model. This common data model has a marked benefit as we transition the EHR from HELP to HELP2. Currently, the clinical data are stored in two completely different databases, the HELP system and the HELP2 system. These systems have different data structures and “dictionaries” (coding systems). Translating the data to a common data model allows the development of rules independent of the data location or structure. Rule developers have no need to know where the data are physically located and/or its structure or codes. This facilitates maintenance of the rules when migrating data from a legacy system to a new platform.⁵⁷

After the rules are executed, the conclusions that are generated are sent to the *dispatch node*. The *dispatch node* is responsible for saving the conclusions to the EHR and delivering them to a destination specified by the user or the rule developer. Currently, the *dispatch node* can send the rules’ conclusion (e.g., alerts, critiques, suggestion, etc.) to pagers, cell phones, email, and to an electronic “in-box” specific to each user.

A *configuration manager* controls the functioning of the all the modules. It is Web-based and allows the system to be managed and configured from any browser. Modules can be configured without having to deactivate the system. The configuration manager also permits the monitoring of salient system functions including error states and performance.

This collection of modules represents an embodiment of the lessons learned from the original CDSS developed over more than two decades for the HELP system. New decision support applications designed and built for HELP2 provide a daily test of our success in learning from our earlier experiences with CDSS.

Intermountain Healthcare's Clinical Knowledge Management Infrastructure

As IHC intensifies the transition from its legacy inpatient information system (the HELP system) to the new component-based clinical information system (HELP2), a new definition of computable medical knowledge has evolved. The examples of CDSS described above were designed to intervene in select medical decisions by providing focused and specific observations or suggestions. The new definition has grown to embrace systems that access collections of more general advice while still respecting the context provided by a selected patient's data and the applications invoked by the user.

The new definition applies to a variety of informational interventions including: (1) tools that reach across the Internet to query commercial and public collections of medical advice, to bring back references appropriate to the medical context surrounding the query; (2) systems that query local collections of problem-specific clinical guidelines to provide context-specific advice on medical care. This advice seeks to promote decisions that are consistent with IHC's care standards; and (3) collections of orders designed to provide a context-specific starting point for clinicians using IHC's new Computer-based Physician Order Entry (CPOE) system.

These broader goals have led to a revised view of the environment required for authoring and maintaining medical knowledge. This view is embodied in a comprehensive clinical knowledge management (CKM) strategy, which is being implemented within the HELP2 computing environment. Below, we briefly discuss this strategy and its implementation. The focus, instead of being on examples, is on the processes of coordinating development and exploitation of computerized medical knowledge and tools to support these processes.

Infrastructure Overview

A key strategy, adopted by Intermountain Healthcare, for promoting consistency and quality in clinical care, is the development and deployment of problem-specific guidelines detailing salient features of that care. These guidelines may be delivered as textual advice suited to the clinical circumstances, or as lists of suggested orders designed to provide an initial order set in a CPOE package.

The strategy for managing the largely descriptive knowledge represented is based on coordinated initiatives that identify and disseminate clinical best practices to help reduce clinical variability and improve disease management processes and outcomes. These initiatives, known as "*Clinical Programs*,"⁵⁸ are developed by interdisciplinary teams supported by specialized workgroups. Development teams and workgroups are recruited from practicing clinicians who provide both domain knowledge and local or regional

representation. A senior physician, recognized as a system-wide domain expert, is commonly the leader of these teams.

In addition to practicing clinicians, each team is also staffed with outcomes analysts, data architects, knowledge engineers, and clinical education professionals. Development teams and workgroups are responsible for the creation of corporate-wide care process models, data collection tools, provider and patient educational materials, clinical documentation templates, and different kinds of computerized decision support elements, such as rules, protocols, care plans, and order sets. These groups are also responsible for the selection and customization of external knowledge sources obtained from public domain sources, or through licensing from commercial vendors.

Content development priorities are established by guidance councils, taking into account the most prevalent and/or variable diagnostic conditions and clinical work processes, complemented by key patient safety processes. The Clinical Programs that have been established so far at IHC cover the following medical specialties and subspecialties: cardiovascular medicine, intensive medicine, neuromusculoskeletal diseases, oncology, pediatrics, preventive care, primary care, surgery, and women and newborns.

Tools to Manage Clinical Knowledge

A complete software infrastructure to support the clinical knowledge management strategy just described has also been developed. The software infrastructure aims at supporting distributed and collaborative processes for authoring, reviewing, and deployment of knowledge content. During the authoring and review phases, all the knowledge content is stored and organized by a *knowledge repository* (KR).

The KR is the cornerstone of the clinical knowledge management software infrastructure. The KR has been implemented using a flexible database model and can be used to store multiple categories of knowledge content, ranging from unstructured narrative text to well structured documents and executable logic modules. Each KR record is considered a *knowledge document* that is preferably represented in XML, but many of the most common *multipurpose internet mail extensions* (MIME) formats are also supported.⁵⁹

Every knowledge document is associated with a *header* XML document that is used to store detailed document *metadata*. The *header* is used to implement the KR's version control mechanism, providing a detailed record of all the changes and enhancements made to any given knowledge document. In terms of searching and retrieving knowledge documents from the KR, a set of specialized services has been created, leveraging existing XML document transformation and presentation standards.⁶⁰ The KR currently provides services to find, retrieve, and/or manipulate the knowledge documents according to the needs of various client applications. However,

content manipulation is limited to instances stored natively as XML documents.

Authoring and review processes for KR documents are supported by two web-based applications: the *Knowledge Authoring Tool* (KAT), and the *Knowledge Review Online* (KRO). KAT is an authoring environment that allows clinical experts to create knowledge documents using XML as the underlying representation formalism.^{61,62} The authoring environment generates XML instances using data entry templates created from document models expressed in XML Schema.⁶² The templates are used to guide and enforce the underlying structure of each knowledge document, implementing a variety of data types that can be used to create simple narrative documents, as well as richly tagged structured documents. The current version of KAT is being used to author 10 different types of documents, ranging from order sets for IHC's CPOE system to corporate nursing care standards.

The main function of KRO is to support an open and distributed review process, where practicing clinicians, i.e., end-users of the knowledge documents, have the opportunity to provide direct feedback to the document authors. The implementation of KRO exposes all the KR knowledge documents to nearly all IHC clinicians through IHC's intranet. Whenever a review is submitted, the author is promptly notified by e-mail. Reviews are also stored in the KR and can be accessed by any other KRO user. Also through KRO, clinicians can subscribe to e-mail alerts that keep them informed about updates and modifications to the documents they have selected. The functions available in KRO are designed to be exposed as simple Web services, enabling users to submit a review or to subscribe to an e-mail alert from within the clinical applications that they routinely use to take care of patients (CPOE, Results Review, etc.).

Application of the Clinical Knowledge Management Infrastructure to Computer-based Physician Order Entry

In the next generation of medical information systems, a fundamental tool for delivering decision support will be a computerized version of the medical order entry system. Both the critiquing and suggestion-based approaches described above are most effective in an environment where the physician personally documents his diagnostic and therapeutic decisions through a direct interaction with the computer. Intermountain Healthcare's approach to implementing CPOE illustrates the use of the knowledge management tools described above.

IHC is in the process of developing a new CPOE system. The CPOE system is a module of the new HELP2 system, and it is being gradually implemented at all IHC's hospitals and outpatient facilities. The CPOE implementation strategy is based on context specific *order sets* as a key factor to encourage physicians' acceptance of the new system.

The development of these order sets utilizes the CKM infrastructure described above, with the underlying assumption that order sets are, in fact,

intervention tools to promote the implementation of clinical care processes that embody best practices and evidence-based guidelines and protocols.⁶³ Once fully implemented, more than 3,500 physicians will be routinely using the new CPOE system.

The effective development of order sets requires a constant collaboration between clinical experts responsible for authoring the order sets and the clinicians who use these sets. Direct and continuous feedback is probably the most efficient mechanism to request fixes or suggest enhancements to the content of the order sets. The dialogue established between authors and users promotes open collaboration and provides a sense of co-ownership of the resulting order sets. IHC considers this process vital for the overall success of the CPOE implementation, and the clinical programs are fully committed to this approach.

Currently, the editorial process for the creation and maintenance of order sets is initiated and controlled exclusively by the lead author. Development teams or workgroups are responsible for nominating the lead authors. Using KAT, the author can create an order set by simply filling the template that has been designed specifically for order sets.⁶¹ Once the authoring phase is completed, the author can publish the order set, so others can review its content and analyze its appropriateness.

As indicated above, the review phase is supported by KRO. Within KRO, every comment and suggestion regarding an order set is instantaneously made available to the author and to the other reviewers. If suggestions made by reviewers require modifications to the order set, the author can make those modifications using KAT and promptly publish a new version of the order set. The authoring and review cycle can be repeated several times, until the content of the order set is considered adequate for clinical use. The approval for clinical use results from the consensus of the group that nominated the lead author. Once the order set is approved, the author is responsible for activating it. The activation is obtained by just changing the status of the order set to "active." At this point, the order set is automatically made available to the CPOE system.

Once order sets are made available to the CPOE system, clinicians begin to use them during the ordering process. In reality, the activation of a brand new order set for clinical use marks the beginning of a secondary review cycle, where authors start receiving feedback from the actual users of the order sets. During this secondary review cycle, the authors are again responsible for analyzing and adopting, or not, the modifications suggested by the users.

At this stage, the most difficult challenge for the author is to try to understand and accommodate the needs of the different CPOE deployment settings. In essence, the lead author, supported by the corresponding development team or workgroup, is directly responsible for making sure the order sets are not only current with published evidence and accreditation requirements, but also reflect and accommodate the peculiarities of the

different clinical settings. All these activities have to be performed in a timely fashion, and in harmony with previously defined best practices. The solution implemented by IHC is based on a collaborative knowledge management approach, where knowledge experts retain the authority to create and modify most of the knowledge content necessary for the CPOE system.

The process used to test and revise CPOE is being put through a series of small prototypes. Select groups of physicians (e.g., teams from the ICUs, groups of surgeons) volunteer to develop order sets and to use the application that allows them to be viewed and modified, and used as the orders for a specific patient. Their experience with this process is used to revise the order sets and the software that delivers them.

The main complaints relate to the absence of a connected order communications system. The physicians create their orders using a computerized tool, but then are required to provide a printed version to the ward clerks for further processing. This extra step will be eliminated when a new order communication system, currently on the drawing boards, is put into service and integrated with the interactive CPOE application.

Summary

In this chapter, we have reviewed a number of hospital-based applications that provide medical decision support. These applications can be categorized in a variety of different ways. We have found it profitable to think of these systems in terms of their relationship to the data, and of their interfaces with their users. These foci should be helpful to future system developers and implementers, as they reflect on the environment required for the success of decision support applications.

We have also attempted to emphasize the range of sophistication that can be found in a clinically operational CDSS. Applications using simple logic can contribute a great deal to the quality of care provided in a clinical setting. Programs that use more complex techniques and that strive to provide the more sophisticated decisions associated with disease recognition can also contribute. Among the diagnostic applications currently functioning in hospital settings, those that focus on specific, limited diagnostic goals with a recognizable target audience have been more successful. General-purpose diagnostic programs, while capable of producing interesting results, have yet to find an audience for which they can provide a routine, valued support function.

The lessons learned from the information systems used in hospitals are diffusing rapidly into the outpatient setting. Less expensive hardware, more flexible software, and an environment that increasingly values the efficiencies that computers can offer are encouraging the development of systems for a wide range of clinical settings. As this process occurs, the lessons gleaned by developers of CDSS systems in a hospital setting provide a

springboard for the decision support systems of the future. These systems will embody, in their software infrastructure, computing models derived from experiments conducted in environments like the HELP system.

As new CDSS systems incorporate the infrastructure and decision models developed in the past, these next-generation systems will also incorporate approaches to knowledge engineering and maintenance that have evolved as a part of the research described above. These knowledge management practices reflect a philosophy of development and continuous review shared by a community of caregivers. Adherence to this approach will do much to reduce the challenges associated with implementing potentially disruptive CDSS technologies, by involving the medical community in their creation and growth.

References

1. Kohn, LT, Corrigan JM, Donaldson MS, eds. *To err is human*. Washington D.C.: National Academy Press; 1999.
2. Bates DW, Leape LL, Cullen DJ, et al. Effect of computerized physician order entry and a team intervention on prevention of serious medication errors. *JAMA* 1998;280:1311–1316.
3. Bates DW, Teich JM, Lee J, et al. The impact of computerized physician order entry on medication error prevention. *J Am Med Inform Assoc* 1999;6:313–321.
4. Hunt DL, Haynes RB, Hanna SE, Smith K. Effects of computer-based clinical decision support systems on physician performance and patient outcomes. *JAMA* 1998;280:1339–1346.
5. Bates DW, Kuperman GJ, Wang S, et al. Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality. *J Am Med Inform Asso.* 2003;10:523–530.
6. Bleich HL. The computer as a consultant. *N Engl J Med* 1971;284:141–147.
7. McDonald CJ. Protocol-based computer reminders, the quality of care and the non-perfectibility of medicine. *N Engl J Med* 1976;295:1351–1355.
8. Tierney WM, Overhage JM, McDonald CJ. Toward electronic records that improve care. *Ann Intern Med* 1995;122:725–726.
9. Clayton PD, Sideli RV, Sengupta S. Open architecture and integrated information at Columbia-Presbyterian Medical Center. *MD Comput* 1992;9:297–303.
10. Safran C, Herrmann F, Rind D, Kowaloff HB, Bleich HL, Slack WV. Computer-based support of clinical decision making. *MD Comput* 1990;9:319–322.
11. Kuperman GJ, Gardner RM, Pryor TA. *HELP: A dynamic hospital information system*. New York: Springer-Verlag; 1991.
12. Teich JM, Geisler MA, Cimmermann DE, Frank AD, Glaser JP. 14th Annual Symposium on Computer Applications in Medical Care; Washington; DC (USA); Nov. 4–7 1990. pp. 735–739.
13. Geissbuhler A, Miller RA. A new approach to the implementation of direct care-provider order entry. *Proc AMIA Annu Fall Symp* 1996:689–693.
14. Pryor TA, Clayton PD, Haug PJ, Wigertz O. Design of a knowledge driven HIS. *Proc Annu Symp Comput Appl Med Care* 1987;11:60–63.
15. Gardner RM, Cannon GH, Morris AH, Olsen KR, Price G. Computerized blood gas interpretation and reporting system. *IEEE Computer* 1975;8:39–45.

16. Gardner RM. Computerized data management and decision making in critical care. *Surg Clin N Am* 1985;65:1041–1051.
17. Hulse RK, Clark SJ, Jackson JC, Warner HR, Gardner RM. Computerized medication monitoring system. *Am J Hosp Pharm* 1976;33:1061–1064.
18. Haug PJ, Gardner RM, Tate KE, et al. Decision support in medicine: examples from the HELP system. *Comput Biomed Res* 1994;27:396–418.
19. Wheeler LA, Brecher G, Sheiner LB. Clinical laboratory use in the evaluation of anemia. *JAMA* 1977;238:2709–2914.
20. Olsen DM, Kane RL, Proctor PH. A controlled trial of multiphasic screening. *N Eng J Med* 1976;294:925–930.
21. Bradshaw KE, Gardner RM, Pryor TA. Development of a computerized laboratory alerting system. *Comput Biomed Res* 1989;22:575–587.
22. Tate KE, Gardner RM, Weaver LK. A computer laboratory alerting system. *MD Comput* 1990;7:296–301.
23. Tate KE, Gardner RM. Computers, quality and the clinical laboratory: A look at critical value reporting. *Proc Annu Symp Comput Appl Med Care* 1993;17:193–197.
24. Tate KE, Gardner RM, Scherting K. Nurses, pagers, and patient-specific criteria: three keys to improved critical value reporting. *Proc Annu Symp Comput Appl Med Care* 1995;19:164–168.
25. Johnson DS, Ranzenberger J, Herbert RD, Gardner RM, Clemmer TP. A computerized alert program for acutely ill patients. *J Nurse Adm* 1980;10:26–35.
26. Kuperman GJ, Teich JM, Bates DW, et al. Detecting alerts, notifying the physician, and offering action items: a comprehensive alerting system. *Proc Annu Symp Comput Appl Med Care* 1996;20:704–708.
27. Gardner RM, Golubjatnikov OK, Laub RM, Jacobson JT, Evans RS. Computer-critiqued blood ordering using the HELP System. *Comput Biomed Res* 1990;23:514–528.
28. Lepage EF, Gardner RM, Laub RM, Golubjatnikov OK. Improving blood transfusion practice: role of a computerized hospital information system. *Transfusion* 1992;32:253–259.
29. Gardner RM, Christiansen PD, Tate KE, Laub MB, Holmes SR. Computerized continuous quality improvement methods used to optimize blood transfusions. *Proc Annu Symp Comput Appl Med Care* 1993;17:166–170.
30. Sittig DF, Pace NL, Gardner RM, Beck E, Morris AH. Implementation of a computerized patient advise system using the HELP clinical information system. *Comput Biomed Res* 1989;22:474–487.
31. East TD, Henderson S, Morris AH, Gardner RM. Implementation issues and challenges for computerized clinical protocols for management of mechanical ventilation in ARDS patients. *Proc Annu Symp Computer Appl Med Care* 1989;13:583–587.
32. Henderson S, East TD, Morris AH, et al. Performance evaluation of computerized clinical protocols for management of arterial hypoxemia in ARDS patients. *Proc Annu Symp Comp Appl Med Care* 1989;13:588–592.
33. Morris AH, Wallace CJ, Menlove RL, et al. A randomized clinical trial of pressure-controlled inverse ratio ventilation and extracorporeal CO₂ removal for adult respiratory distress syndrome. *Am J Respir Crit Care Med* 1994;149:295–305.

34. Johnson JA, Bootman HL. Drug-related morbidity and mortality: a cost of illness model. *Arch Intern Med* 1995;155:1949–1956.
35. Classen DC, Pestotnik SL, Evans RS, Burke JP. Computerized surveillance of adverse drug events in hospital patients. *JAMA* 1991;266:2847–2851.
36. Evans RS, Pestotnik SL, Classen DC, et al. Development of a computerized adverse drug event monitor. *Proc Annu Symp Comput Appl Med Care* 1991: 23–27.
37. Naranjo CA, Busto U, Sellers EM, et al. A method for estimating the probability of adverse drug reactions. *Clin Pharmacol Ther* 1981;30:239–245.
38. Classen DC, Pestotnik SL, Evans RS, Llyod JF, Burke JP. Adverse drug events in hospitalized patients: excess length of stay, extra costs, and attributable mortality. *JAMA* 1997;277:301–306.
39. Evans RS, Larsen RA, Burke JP, et al. Computer surveillance of hospital-acquired infections and antibiotic use. *JAMA* 1986;256:1007–1011.
40. Evans RS, Burke JP, Pestotnik SL, Classen DC, Menlove RL, Gardner RM. Prediction of hospital infections and selection of antibiotics using an automated hospital data base. *Proc Annu Symp Comput Appl Med Care* 1990:663–667.
41. Evans RS, Classen DC, Pestotnik SL, Clemmer TP, Weaver LK, Burke JP. A decision support tool for antibiotic therapy. *Proc Annu Symp Comput Appl Med Care* 1995:651–655.
42. Evans RS, Pestotnik SL, Classen DC, et al. A computer-assisted management program for antibiotics and other antiinfective agents. *N Engl J Med* 1998;338: 232–238.
43. Ledley RS, Lusted LB. Reasoning foundations of medical diagnosis. *Science* 1959;130:9–21.
44. Warner HR, Rutherford BD, Houtchens B. A sequential Bayesian approach to history taking and diagnosis. *Comput Biomed Res* 1972;5:256–262.
45. Haug PJ, Warner HR, Clayton PD, et al. A decision-driven system to collect the patient history. *Comput Biomed Res* 1987;20:193–207.
46. Haug PJ, Rowe KG, Rich T, et al. A comparison of computer-administered histories. *Proc Am Assoc Med Syst Inf Annu Conf* 1988:21–25.
47. Herman PG, Gerson DE, Hessel SJ, et al. Disagreements in chest roentgen interpretation. *Chest* 1975;68:278–282.
48. Yerushalmy J. Reliability of chest radiology in the diagnosis of pulmonary lesions. *Am J Surg* 1955;89:231–240.
49. Koran LM. The reliability of clinical methods, data, and judgments (second of two parts). *N Engl J Med* 1975;293:695–701.
50. Rhea JT, Potsaid MS, DeLuca SA. Errors of interpretation as elicited by a quality audit of an emergency radiology facility. *Radiology* 1979;132:277–280.
51. Raines CJ, McFarlane DV, Wall C. Audit procedures in the national breast screening study: mammography interpretation. *J Can Assoc Radiol* 1986;37: 256–260.
52. Shannon CE, Weaver W. *The mathematical theory of communication*. Urbana: University of Illinois Press; 1949.
53. Haug PJ, Clayton PD, Tocino I, et al. Chest radiography: a tool for the audit of report quality. *Radiology* 1991;180:271–276.
54. Haug PJ, Pryor TA, Frederick PR. Integrating radiology and hospital information systems: the advantage of shared data. *Proc Annu Symp Comput Appl Med Care* 1992:187–191.

55. Haug PJ, Rocha BH, Evans RS. Decision support in medicine: lessons from the HELP system. *Int J Med Inform* 2003;69:273–284.
56. Clayton PD, Narus SP, Huff SM, et al. Building a comprehensive clinical information system from components. The approach at Intermountain Health Care. *Methods Inf Med* 2003;42:1–7.
57. Pryor TA, Hripsak G. Sharing MLM's: an experiment between Columbia-Presbyterian and LDS Hospital. In: Safran C, ed. *Proceedings of the Seventeenth Annual Symposium on Computer Applications in Medical Care*. New York: McGraw-Hill; 1993:399–404.
58. IHC Clinical Programs. www.intermountainhealthcare.org/xp/public/physician/clinicalprograms/womennewborns/circumcision.xml; accessed July 2006.
59. W3C Architecture domain. The Extensible Stylesheet Language Family (XSL). www.w3.org/Style/XSL/; accessed July 2006.
60. Hulse NC, Rocha RA, Bradshaw R, Del Fiol G, Roemer L. Application of an XML-based document framework to knowledge content authoring and clinical information system development. *Proc. AMIA Symp* 2003:870.
61. Hulse NC, Rocha RA, Del Fiol G, Bradshaw RL, Hanna TP, Roemer LK. KAT: a flexible XML-based knowledge authoring environment. *J Am Med Inform Assoc*. 2005;12:418–430.
62. Sperberg-McQueen CM, Thompson H. XML Schema. www.w3.org/XML/Schema; accessed July 2006.
63. Del Fiol G, Rocha RA, Bradshaw RL, Hulse NC, Roemer LK. An XML model that enables the development of complex order sets by clinical experts. *IEEE Trans Inf Technol Biomed*. 2005;9:216–228.

9

Clinical Decision Support Within the Regenstrief Medical Record System

BURKE W. MAMLIN, J. MARC OVERHAGE, WILLIAM TIERNEY,
PAUL DEXTER, and CLEMENT J. McDONALD

In this chapter, we review a timeline of Regenstrief Medical Record System (RMRS) studies related to decision support, describe the system's architecture, and discuss the lessons we have learned from over a quarter century of experience with clinical decision support systems (CDSS). From the beginning, the RMRS has included mechanisms for writing rules for generating reminders to physicians, based on clinical data, including laboratory results, visit diagnosis, coded medications prescribed in the clinic, and vital signs collected on encounter forms.

Reminders and Guidelines—A History of Regenstrief Medical Record System Decision Support*

The Regenstrief Institute (Indianapolis, IN) began developing the RMRS in 1972. It was first installed in the diabetes clinic of Wishard Memorial Hospital (Indianapolis, IN), running on a Digital Equipment Corporation PDP 11/44 computer with four access lines.² In 1974, the use of this system was expanded to a few of the hospital's many general medicine clinics.

In 1976, we published our first randomized, controlled trial of reminders in the *Annals of Internal Medicine*.³ In this and most of our subsequent reminder studies, we were motivated by the fact that humans are imperfect information processors, subject to oversights and distractions. We introduced this study based on the following premises, providing a cogent rationale for this and all of our subsequent reminder studies:

There is a general tendency to associate the process of medical care with sophisticated intellectual activity—the creation of causal hypotheses from knowledge of biomedical mechanisms, the choice of therapeutics based on summed experi-

* An earlier review of computerized reminders at Regenstrief Institute was written by Prokosch and McDonald in 1995.¹

ence. It is easy to forget that at the working level much medical activity consists of simple recognition-response arcs—the hematocrit fall triggers a test of the stool for occult blood; the newly positive purified protein derivative test stimulates a chest X-ray. For such “reflexes,” the stimulus is the occurrence of a simple clinical event, the response is a corrective or clarifying action, and the connecting logic is defined by simple rules or protocols. In these terms, much of routine (but important) medical activity requires compulsive attention to protocol rather than intellectual brilliance.

On occasion, physicians’ clinical “reflexes” fail, as evidenced by errors with respect to the management of simple clinical events. Cole, Balmer, and Wilson⁴ found delay or omission of the proper treatment in 8 of 30 cases of tuberculosis. Three of these involved failure to note positive culture reports shown on the chart. Others have found similar errors of attention with respect to X-ray reports of active tuberculosis.^{5,6} Greenblatt and Koch-Weser⁷ noted that 77 of 781 patients on spironolactone were being given supplemental potassium chloride (KCl) despite blood urea nitrogen (BUN) levels greater than 25 mg/dl, and that 41% of these were hyperkalemic. In a report by Shapiro, Slone, and Lewis,⁸ respiratory depressants caused the death of 2 patients with chronic lung disease. Caranasos, Stewart, and Cluff⁹ reported that 2.9% of hospitalizations are the direct result of the effects of medication. Among 606 patients from a general medicine clinic, Kelley and Mamlin¹⁰ found 7 with cervical carcinomas, 38 with intraocular hypertension, 17 with untreated congestive heart failure, 9 with recent myocardial infarcts and numerous patients with abnormal chemical test results, all of whom had their diseases go undetected despite regular clinic visits.

Can errors such as the above be reduced by prospective, protocol-driven computer reminders about the existence and implications of simple clinical events?

This study, which was performed in the Wishard diabetes clinic, used pre-specified rules to identify patients who were eligible for particular clinical actions and then generated reminders to physicians about such actions. For example, if a patient was taking aminophylline and his or her serum aminophylline level had not been measured within a designated time period, the computer generated a reminder to the physician about ordering an aminophylline level. For this study, the reminder rules were executed each night as a batch process that examined the computer records of patients who were scheduled for visits on the next day. The reminders for each patient were delivered as a paper report that the clinic staff attached to the front of the patient’s chart, along with a computer-generated flow chart and encounter form. The flow chart displayed the patient’s active drug profile and provided space for writing notes and orders. The study was a randomized, controlled trial, and physicians were randomized to intervention status (received reminders) or control status (did not receive reminders). Patients inherited the study status of their respective physicians. We developed nearly 300 rules in two categories:¹¹

1. Reminders to order a specific test because a particular medication had been prescribed for the patient. The rules were of this general form: “if on drug A and no test B for X months, then order test B.”

2. Reminders to change therapy in response to an abnormal measurement. The general form of these rules was: "if on drug A and the first dose was followed by a change in test B for the worse, then warn about a possible causal relation."

The computer generated reminders for both intervention and control patients, but delivered them only to the physicians in the intervention group. The primary outcome was the rate at which physicians responded to the computer-suggested actions for the eligible patients.

During the 8-month study, 601 visits (301 visits by 119 study patients and 300 visits by 107 control patients) activated one or more study rules. Sixty-three clinicians responded to 36% of events with reminders and 11% without ($p < 0.0001$). They responded to 28% of second-category events with reminders and 13% without ($p < 0.026$). When only the most clinically significant events (e.g., increase the antihypertensive regimen for diastolic blood pressure greater than 110 mm Hg) were considered, the clinicians responded to 47% of events when reminded, but only 4% when not. This study showed that computer-generated reminders could improve clinical processes, and it was the first randomized, controlled study of computer-based decision support to show a significant effect on the care process.

In a subsequent study published in the *New England Journal of Medicine*, McDonald described the "nonperfectibility of man."³ This study, performed in the general medicine clinic (not the diabetes clinic), included more reminder rules (390 compared to 300) and used a cross-over design, in which physicians served as their own controls (i.e., physicians were the intervention subjects in one phase of the study and controls in another). In this study, we followed 9 physicians practicing in a general medicine clinic for one half-day per week for 17 weeks. During the intervention phase, clinicians were asked to initialize the reminder reports to indicate that they had actually seen the reminders. This study included three kinds of reminders:

1. Type I: suggestions to observe a physical finding or inquire about a symptom;
2. Type II: suggestions to order a diagnostic study;
3. Type III: suggestions to change or initiate a therapeutic regimen.

The reminders were printed on the encounter form that clinicians used for writing notes and orders, as well as on the one-page reminder sheet. Physicians acted on the computer-suggested actions in 51% of 327 intervention events and 22% of 385 control events ($p < 0.00001$). The rate at which they responded was not higher for physicians whose control periods followed their study periods; thus the computer-generated suggestions had no "training effect." In other words, the computer reminders did not teach the physicians something they did not know, but rather activated the physicians' pre-existing knowledge and intentions.

Interestingly, during the study, one medicine resident insisted, “What was needed was education of bad physicians about what they did not know, not reminders to good physicians about what they knew.” Because each physician served as his or her own control, we could observe the effect of the reminders on each individual physician. As it turned out, this particular physician’s response to reminders was larger than that of any other study participant. He took the suggested action in 75% of the cases when he was reminded, but only in 25% of the cases when not.

A subsequent study, which used 410 computerized protocols and studied 31 physicians over 17 weeks, corroborated the findings of the first two studies.¹² In this 1980 study, we offered access to the medical literature that supported the reminders, in addition to an improved and more sophisticated set of reminder rules.

As part of this third study, we stationed a pharmacy faculty member in the clinic, with copies of reference articles that provided justifications for many of the reminders. The reminder reports given to care providers cited these references and invited the providers to obtain original copies from the nearby pharmacist. Physicians responded significantly more often with reminders than without; indeed, reminders had a two-fold or greater effect on practitioner behavior. However, the reminders evoked little physician interest in the supporting literature. In fact, not a single cited article was requested from the pharmacist during the study. The physicians did not ask for copies of the referenced articles (given by house officers in informal discussions) because of: (1) the time pressure of patients waiting to be seen back on the wards; and (2) the fact that they agreed with, and knew the evidence that justified, the reminder (i.e., they did not need convincing).

By the late 1970s, the RMRS had grown into a full, electronic medical record repository for Wishard Memorial Hospital, the reminder rules had been compiled into a formal computer language (CARE), and the scope of the rules had increased substantially. During 1978–1980, we performed our largest reminder study—a two-year, randomized, controlled trial that included 1,490 different reminder rules, the complete set of which were published.¹³ This two-year study included 130 different providers (including house officers, nurse practitioners, and faculty members), nearly 14,000 patients, of whom 90% were eligible for one of the reminder actions, and more than 50,000 visits. During this study, the computer generated 140,000 reminders for both intervention and control visits.¹⁴ This was the largest randomized controlled trial (RCT) of a reminder system ever performed.

In the 1978 study, we used a more sophisticated randomization (by practice—called “teams” in our environment—rather than by individual providers) to minimize the possibility of contamination between intervention and control physicians. We also used more sophisticated analytic techniques¹⁵ to take into account possible clustering effects of patients within providers and/or providers within study teams. We used the same general strategy of delivering reminders on paper reports that we had used in all

previous studies. During the study, the computer suggested an average of six different clinical actions per patient (e.g., testing for occult blood or adding a beta-blocker to the patient's treatment regimen). The reminders increased the physicians' completion of the suggested actions from 29% of the eligible control patients to 49% of the eligible intervention patients. The 15 most frequently suggested clinical actions accounted for 69.8% of all reminders given. The reminders had the greatest effect on preventive care (e.g., influenza vaccination, pneumonia vaccination, occult blood testing), for which reminders increased the usage rate by two- to four-fold!

In all of the studies discussed thus far, intervention physicians took the suggested actions for a much larger percent of eligible patients than did control physicians. However, even with reminders, physicians failed to take the suggested actions for 40–50% of the eligible patients. Some of this lack of response can be explained by the fact that the physicians did not always see the reminder reports. In one of the studies, we asked physicians to initial all reminder reports they had seen or read (whether or not they agreed with the reminder). This allowed us to distinguish between reminders seen by the physician and those that were not seen. We discovered that an important component of the variance in response rate (15%) was explained by the physician not seeing or reading the reminders. We believe there is room to improve the physicians' rate of responding to the reminder reports. Nevertheless, we should not expect to get to a 100% response rate because: (1) in some cases, a nonresponse is the correct response; (2) the patient will not always accept the suggested action; and (3) the computer does not always have all of the relevant information about the suggested action. After auditing a sample of the charts,¹⁴ we found gaps in the computer's patient information base that would have invalidated the reminders in 5% (pneumonia vaccine) to 50% (cervical pap testing) of cases, depending upon the action suggested.

We also assessed whether it was necessary to deliver reminders in real time. Could similar results be obtained by providing feedback reports to physicians, indicating which patients they had seen in the previous month who were eligible for preventive care but had not received it? We compared such feedback reports with real-time computer reminders in a 2×2 factorial randomized, controlled trial.¹⁶ This 7-month study included 6,045 patients and targeted 13 preventive care protocols, both testing (e.g., mammograms) and treatment (e.g., oral calcium carbonate for osteoporosis prophylaxis). Each protocol could be delivered as a monthly feedback report or as a printed reminder in the clinic (immediate feedback). Suggestions on the feedback reports were followed by specific actions the physician could take: (1) reschedule the patient sooner; (2) perform the test during the next scheduled visit; (3) stop the reminder for this patient (not applicable, but the physician agrees, in principle, with the protocol); (4) stop the reminder for this patient (physician disagrees with protocol); and (5) pull patient's chart for review. Both reminders and feedback reports increased adherence

to the preventive care suggestions, although reminders were significantly more powerful than feedback reports. Notably, the combination of reminders and feedback reports was no better than feedback reports alone. This is understandable, since the physicians checked “remind me next visit” for 80% of the suggestions on the feedback reports.

Following the successful demonstration that computer reminders substantially increased adherence to practice guidelines, especially those targeting preventive care, computer reminders were implemented throughout Wishard’s general medicine and primary care pediatric clinics. Consistent with the process of continuous quality improvement,¹⁷ we then sought ways to further increase physicians’ adherence to guideline-based care suggestions. In one study, we assessed the effects of forcing physicians to respond to selected reminders for cancer screening tests.¹⁸ In a randomized, controlled trial, intervention physicians had to either order the suggested test (cervical Papanicolaou smear, mammogram, or fecal occult blood test) or explain why they did not do so. This study included 145 physicians and 5,407 patients. Intervention physicians followed 46% of the reminders compared with 38% for control physicians ($p = 0.002$). Intervention physicians’ reasons for not adhering to the guidelines included the physician being too busy or the patient being too sick that day (23%), the reminder being inappropriate (23%, mostly due to missing data on prior hysterectomies), and the patient’s refusal to take the test (10%).¹⁹

In this study, we demonstrated large effects on care processes, and, with a secondary analysis, we showed a large reduction in winter morbidity. This effect was limited to patients who were eligible for influenza vaccines in the years when a large excess in pneumonia mortality occurred.²⁰ Nonetheless, we should not expect to be able to demonstrate improved outcomes in every process intervention, because individual small scale process interventions often do not have the huge sample sizes that would be needed to show outcome effects. The HIP Mammography study included nearly 60,000 eligible women and followed them for over 7 years.²¹ Our reminder study included less than 3,000 eligible women and followed them for 2 years.

During the 1980s, as the RMRS grew, we developed and implemented a physician workstation—the Medical Gopher²²—with physician order entry (CPOE) and note writing capabilities. This system was implemented using personal computers networked to the RMRS. The Medical Gopher benefited from the existing electronic medical record, while enabling direct physician interaction for real-time decision support.^{22,23} While CPOE can exist apart from CDSS, most CPOE systems include aspects of CDSS. In our case, we integrated decision support into the Medical Gopher from its inception and leveraged the direct physician interaction to facilitate further studies.

We used the Medical Gopher order-entry system in the outpatient setting to study the effects of three different interventions in laboratory

test ordering: the display of previous test results,²⁴ charges for tests,²⁵ and/or the likelihood of a positive result when tests were being ordered, respectively.^{18,24–26}

The first study included 111 internal medicine physicians in an outpatient medicine clinic and ran for 16 weeks. Scheduled patients were randomized to intervention or control status, and previous test results were displayed when any of a selected subset of the most common test panels (e.g., CBC, electrolyte, Chem 12) was ordered on intervention patients. When the computer displayed previous test results for commonly ordered tests, the physicians ordered 16.8% fewer of such tests and generated 13% lower test charges.²⁴

In another study, William Tierney, MD, developed logistic regression equations based on data in the RMRS about medication use, previous test values, and demographics to predict the likelihood that a test result would be abnormal. Logistic regression equations were developed for 8 different laboratory tests. When the computer displayed the predicted probabilities that a test would be abnormal—most of which were much lower than the physicians expected—physicians ordered significantly fewer tests, resulting in a 9% reduction in charges.²⁶

Displaying the charges for individual outpatient tests, along with the total charge for all tests ordered during the outpatient visit, also reduced test ordering.²⁵ We studied 121 physicians' order entries in an outpatient medicine clinic over 59 weeks. Half of the physicians were randomized to the intervention group and the other half to the control group. For 14 weeks before the intervention began, the test ordering rates and related charges were the same across both groups. During the 26-week intervention, when the computer displayed the charge of each new test ordered and the cumulative charges for all tests ordered during that session, physicians ordered 14% fewer tests per patient visit and generated charges that were 13% lower (\$6.68 per visit). The effect was greater for scheduled visits than non-scheduled visits and fell back to near baseline in the 6 months following the intervention. No change in adverse outcomes (hospitalizations, ER visits, and clinic visits) occurred in the intervention group.

During the late 1980s, we extended the use of the Medical Gopher from the outpatient clinics into the inpatient wards of Wishard Memorial Hospital, and we performed the first randomized study of CPOE compared to traditional, paper order entry on the inpatient medicine service. In this study, the order-entry system provided problem-specific menus, order-specific templates, a display of the patient's charge for each item, and warnings for allergies, drug-drug, or drug-diagnosis interactions. The menus and templates were designed to encourage cost-effective ordering and discourage expensive treatments. We did not include active reminders in this study. Six inpatient medicine ward services were randomized to intervention or control groups.

Over 19 months, 68 teams, each comprised of a faculty member, residents, and medical students, were randomly assigned to these services and cared for 5,219 patients. We accounted for house staff returning for additional rotations by placing them onto a service with the same status—intervention or control—as their initial rotation. Physicians and students on ward services randomized to the intervention arm used the PC-based order-entry system for all orders, while control services continued to use a traditional paper-based, order-writing chart and had no access to the Gopher system. The patients seen on intervention services were not significantly different from those on control services. However, we demonstrated a savings of 12.7% (nearly \$900) per admission when the CPOE system was used.²⁷ Further, hospital stays for intervention admissions were 10.5% (0.89 days) shorter than controls ($p = 0.11$). No differences in hospital re-admissions, ER visits, or clinic visits were seen at 1 and 3 months following discharge. When the cost savings were extrapolated to all of the hospital's medical services, the predicted savings were \$3 million per year.

Once the Medical Gopher order-entry system was in regular use in both inpatient and outpatient settings, we began using it as a platform for real-time, complex decision support. We created a new, rule-writing language for the Gopher order-entry system that we called G-CARE (short for Gopher-CARE). Physicians can use reminders generated from G-CARE for real-time responses to data and orders entered as part of the order-entry process, while also considering data already stored in the RMRS.²³ When a G-CARE rule executes, it produces a multivalued result that may include a date, an identifier, a numeric value, a logical value, and/or a prespecified order. Rule designers can write six types of rules: (1) *algebraic* rules for evaluating expressions; (2) *logical* rules to execute a logical operation and return a value of true or false; (3) *prompt* rules to request input from the user; (4) *reminder* rules that returns one set of possible reminder texts and/or order sets; (5) *selection* rules to retrieve data from other G-CARE rules or the medical record; and (6) *special* rules to execute special system code. These rules can be activated at many steps in the order-entry process, including at the start of an order session (before the physician enters any data), immediately after the physician enters a medical problem, at the time the physician selects an order for processing, when the physician completes an order, and at the completion of an order session. These alternatives are further described elsewhere.²³ G-CARE can be used to provide prior test results, suggest orders for baseline testing or follow-up monitoring, or block contraindicated orders and suggest alternatives. For example, the computer might suggest a nuclear medicine renal study instead of an intravenous pyelogram (IVP) in a patient with renal insufficiency.

Since we had so much success with preventive care reminders in the outpatient setting, we expected the same success in a study of preventive care

reminders on the inpatient side. However, our initial attempt was disappointing; our first inpatient reminder study was a negative study.²⁸ We attributed these negative findings to two factors: (1) physicians think of preventive care (e.g., immunization) as belonging in the outpatient, not the inpatient, side of patient care; and (2) the delivery method for reminders was too timid and indirect—a message advised the physician that reminders were available and required him/her to press a special key to see the reminders.

In a more recent inpatient study that focused on 4 preventive measures (pneumococcal vaccination, influenza vaccination, aspirin for vascular disease risk, and subcutaneous heparin for deep vein thrombosis (DVT) prophylaxis), real-time reminders to inpatient physicians *did* have the large effect we expected.²⁹ In this study, providers were randomized into intervention and control groups, and intervention physicians received reminders as orderable pop-ups. Over half of the 6,371 patients admitted to a general medicine service over an 18-month period were eligible for 1 or more of the 4 preventive measures. Their ordering rates (intervention versus control) were 35.8% versus 0.8% for pneumococcal vaccination, 51.4% versus 1.0% for influenza vaccination, 32.2% versus 18.9% for prophylactic heparin, and 36.4% versus 27.6% for prophylactic aspirin ($p < 0.001$ in all cases). We attributed the success of this later study to small—but important—changes in the delivery of the reminders to the ordering physician. Unlike in the previous study, reminders popped up as an order that the physician could accept or reject, but which they could not ignore.

We also have shown the capability of real-time, computer-generated reminders to decrease errors of omission in regard to monitoring therapies and disease in the hospital setting.³⁰ Overhage developed a set of orders that were required as corollaries to orders for 87 selected test and treatment orders. For example, an order for a heparin drip would have an order for an activated partial thromboplastin time (APTT) measurement to follow the effect of the heparin as a corollary. Over a 6-month trial, physicians who had been given reminders about corollary orders ordered them for 46.3% of their eligible patients compared to a 21.9% ordering rate under comparable circumstances by control physicians ($p < 0.0001$). Pharmacists intervened for errors considered to be life threatening, severe, or significant 33% less often for intervention physicians than for control physicians. No significant change in length of stay or hospital charges was detected.

We studied the effects of reminders to physicians to discuss advanced directives with their patients who were either among the oldest old (≥ 75 years) or had serious chronic conditions. In a 2×2 factorial design, we reminded physicians to either discuss instruction directives (instructions about specific types of care—cardiopulmonary resuscitation, ventilation, surgery, artificial hydration and nutrition, etc.—and whether the patient would want the care if terminally ill), proxy directives (establishing durable power of attorney for health care), both instruction and

proxy directives, or neither advance directive (controls). Both types of reminders increased the rate at which physicians discussed advanced directives with eligible patients, with the greatest effect—a six-fold increase—observed for physicians who received both types of reminders over the one-year study period (4% for controls compared to 24% for these physicians).³¹

As guidelines were introduced into the order-entry system using G-CARE, we gained some valuable insights.³²⁻³⁴ Tierney and colleagues described some of the problems encountered when developing computerized guidelines for heart failure. For example, information in published guidelines can be horribly inadequate for defining a computer-executable reminder rule. Published guidelines often include vague terminology, omit branch points, rely on data not available in the electronic records, and fail to address comorbidities or concurrent therapy.

Despite these limitations, using G-CARE we were able to program into the outpatient Gopher workstations detailed guidelines for managing ischemic heart disease and heart failure, hypertension, and reactive airways disease. In a 2-year randomized, controlled trial, 2,123 patients were prospectively enrolled, with 700 having each of the above conditions. Half of the physicians were randomized to receive suggested orders for the management of these chronic conditions, while half did not. All physicians used Gopher to write all orders. A retrospective review of 10% of the charts of patients included in this study showed that the suggested care was indeed indicated. However, receiving suggested orders had no effect on adherence to any of the evidence-based suggestions, clinical outcomes (e.g., hospitalizations or emergency department visits for heart or lung disease, blood pressure control), or health-related quality of life for patients with heart disease,³⁵ lung disease,³⁶ or hypertension.³⁷ These same physicians, who adhered to preventive care reminders, ignored most reminders about chronic disease management. Querying them on their responses to guidelines for managing chronic conditions, we found them to have mixed feelings. Although they found the guidelines to be good sources of information, they also felt they were intrusive and mostly aimed to control costs. The management of chronic disease is also more complicated and subject to more special cases and alternatives than can be easily incorporated into computer reminder rules.

Beyond Reminders—Other Uses of Our General Decision Logic

We have applied the general capabilities of the decision logic for many purposes that we have not formally studied or reported. Most of these applications have been developed in response to cost, administrative, or regulatory needs.

Saving Costs and Improving Efficiency Through Reminders

We use CARE rules to suggest less expensive medications and therapies when the physicians choose the most expensive choices in a given therapeutic class. For example, when physicians order one of the patented and expensive SSRI antidepressants, we direct their attention to fluoxetine (Prozac), which is roughly one fifth the cost (because it has gone off patent) but equally efficacious (see Figure 9.1).³⁸ In the case of fluidized beds, we remind providers of the option of a static air mattress, which some studies have shown to be as good but at a much lower cost. Of course, the computer also enforces formulary controls on medication prescribing.



FIGURE 9.1. Suggestion after ordering an expensive SSRI antidepressant.

We also have rules and logic that help the hospital physicians adhere to Medicare regulations about short-stay patients. For example, the computer reminds physicians to correctly classify their patients as short-stay patients at admission, and then shortly before the 36-hour limit, the computer reminds the responsible physician(s) to write full admissions orders if necessary.

Assistance to Clinical Research

After IRB approval of a given clinical trial, investigators generally must get the approval of the physician who is responsible for a given patient before inviting that patient into the clinical trial. We use Gopher reminders to help in this recruitment process. In the outpatient clinics and emergency rooms, an order session requires the physician to enter the medical problem that was the focus of the patient's visit. Reminder rules linked to the visit diagnosis entry can check for inclusion or exclusion criteria and, when the patient meets criteria, ask the physician whether the patient should be invited into the study. The system can then immediately page the study nurse coordinator to invite approved patients into the study while they are still on site.

Other Decision-Support Mechanisms

The Gopher system does not use the generalized (G-CARE) rules engine to deliver *all* of its decision support. It uses specialized programs driven by tables to deliver decision support for specialized purposes—such as drug interactions, formulary management, and medical necessity checking.

Drug-Drug, Drug-Problem, and Drug-Allergy Checking

Drug-drug, drug-problem, and drug-allergy checks are evoked when a treatment order is entered. The computer uses a table that maps a specific drug to its drug classes. For example: trimethoprim-sulfamethoxazole is a *sulfonamide* and a *triamterene* because it is a combination drug; and metoprolol is both a *beta blocker* and a *selective beta blocker*. The computer also keeps a list of the classes to which each active drug order belongs, as well as the class of problems to which each problem in the active problem list belongs. Drug-drug interactions are checked at the class level by examining all combinations of the entered drugs' classes with the patients' currently active drugs' classes. For each interacting pair, the computer displays a window alerting the physician to the interaction and explaining its importance (see Figures 9.2 and 9.3). Our interaction table includes only the strong and scientifically well supported drug interactions, so this processing occurs at blink speed. The physician's workflow is interrupted only by important interactions.

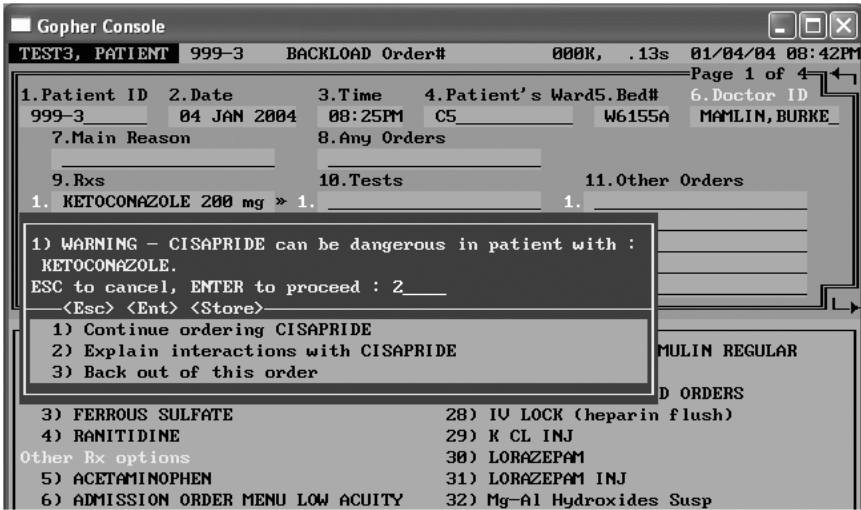


FIGURE 9.2. Drug-drug interaction.

Drug-problem interactions and allergy testing use the same general approach; all checking is done at the class level.

Insurance-Based Formulary Control of Medication Prescribing

We have specialized programs that can check on insurance plan-dependent formularies, displaying the alternatives within a given drug class, and their

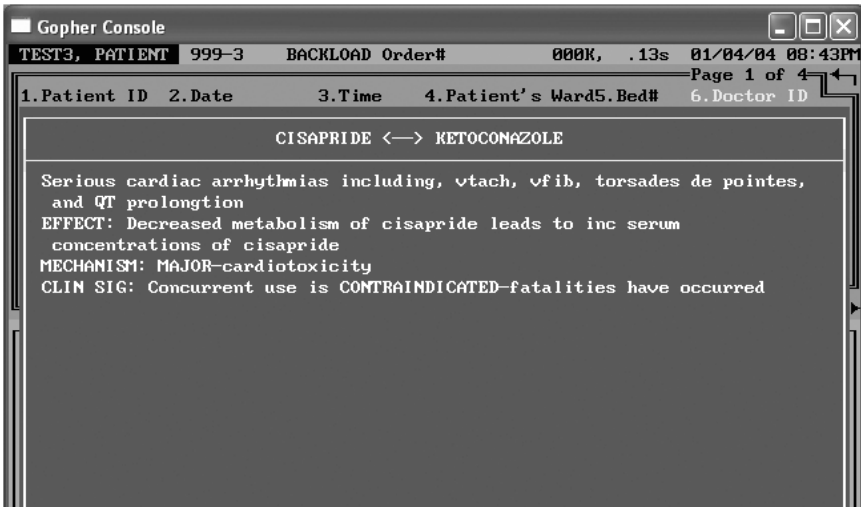


FIGURE 9.3. Drug-drug interaction details.

relative costs. So, if a physician enters an order for the drug Nexium,[®] a proton pump inhibitor, a menu will pop up with the proton pump inhibitor(s) that the patient's insurance plan will cover. The drug costs are indicated by the number of dollar signs shown to the left of the drug name (ranging from "\$" for low cost to "\$\$\$\$" for very high cost). The lists of drugs that are provided within a given class and their costs vary by insurance plan. For example, for one plan, Prilosec[®] (omeprazole) may be the only drug, while another plan might include Nexium[®] (esomeprazole) because of special contract arrangements between the insurance plan and the pharmaceutical manufacturer. Formularies differ from one insurance plan to the next, and individual plans change their formularies over time. Because keeping track of all of these details is maddeningly difficult, such formulary-specific content has great value for physicians.

However, we have been able to deploy this capability for only two of our hospital's major insurance carriers because insurance plans rarely provide their formularies in electronic form, thus requiring that they be updated manually. Standards in this area are desperately needed.

Medical Necessity

For a subset of diagnostic tests, Medicare mandates that testing meet their requirements for medical necessity. The Medicare intermediaries provide tables with a list of ICD-9 diagnostic codes for each of the medical necessity tests. When one of the qualifying diagnoses appears in the diagnosis field for a patient's visit, Medicare accepts the test as medically necessary. When a visit includes a test that requires medical necessity checking and the billing diagnosis does not include one of the qualifying diagnoses for the ordered test, Medicare requires the practice to have the patient sign a special document stating that he or she will pay for the test. It is important to note that many kinds of tests and procedures that are actually medically necessary are not classified as such under the Medicare rubric. This medical necessity requirement places new burdens on medical practice; it requires that the check-out process include the CPT codes for all tests ordered and a computer system that checks to determine whether medical necessity is met for each of those tests. The practice also is put in the awkward position of having to state that the tests are not medically necessary (by Centers for Medicare and Medicaid Services' rules), even when they may be quite necessary by the standards of usual medical care.

In the Gopher order-entry system, we alleviate this problem by asking providers to identify a reason for each test they enter. They can choose from the patient's active problem list and/or from a list of common reasons for ordering the test. When a test does not meet medical necessity by Medicare's standards, the computer informs the physician, so that he or she can either explain it to the patient or reconsider ordering the test. This puts a small, additional burden on the physician, but it smoothes the check-out

process and reduces the number of situations in which the patient is surprised with additional charges.

Passive Billboards, Menus, and Fill-in-the-Blank Templates to Guide Decisions

The Gopher system also provides decision support through static text displays (that we call “billboards”), as well as context-specific menus.

When a provider enters the name of an order, the computer pops up a window with up to eight lines of text to inform (or guide) the provider about that order (see Figure 9.4).

We call this text the billboard because it is designed for quick reading and is focused on the order being written. We use this order-specific billboard to warn physicians about the costs and the most important dangers/benefits of the tests/treatments being ordered. In some cases, the billboard text will suggest alternatives. Using G-CARE rules, we can make portions of the billboard text dynamic. For example, we can include the date that a test was last ordered and its result or embed clinical calculations (e.g., body mass index or creatinine clearance) into the text.

The same window (just beneath the billboard) provides areas for entering the values needed to complete an order or a note. Such details include dose, amounts to dispense, instructions, and reasons for the order. Most of these values can be entered using order-specific menus. The content of these menus also provides guidance to the order-entry process. The most important field required to complete an order is the Sig. (i.e., instruction

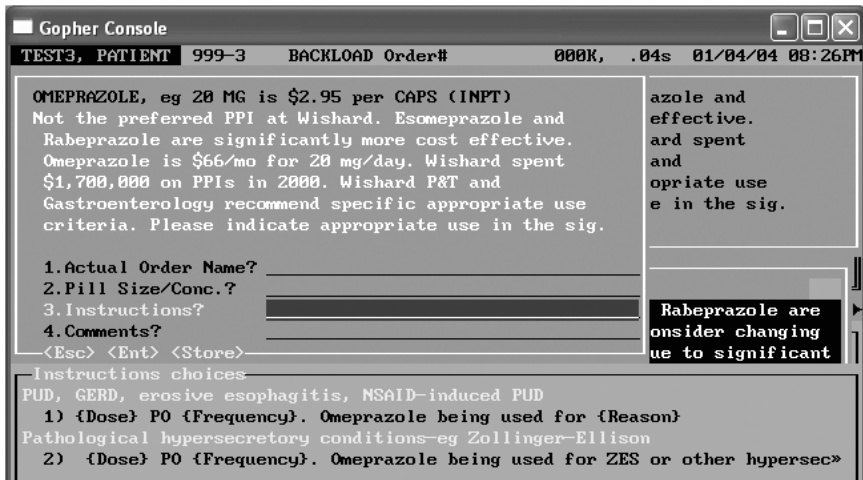


FIGURE 9.4. Billboard shown during a medication order.

Gopher Console

TEST3, PATIENT 999-3 BACKLOAD Order# 000K, .22s 01/02/04 09:07PM Page 1 of 4

1. Patient ID	2. Date	3. Time	4. Patient's Ward	5. Bed#	6. Doctor ID
999-3	31 DEC 2003	02:09PM	C5	W6155A	MAMLIN, BURKE
7. Main Reason	8. Any Orders				
9. Rxs			10. Tests		11. Other Orders

TMP-SMX DS

1. Pill Size/Conc.?
 2. Instructions?
 3. Comments?
 <Esc> <Ent> <Store>

Instructions choices

UTIs - RX

1) 1 tab PO Q12H X {duration}

UTI prophylaxis

2) 1/2 tab PO HS X {duration}

Prostatitis

3) 1 tab PO Q12H X {duration}

Enteritis

4) 1 tab PO Q12H X 7 days

Bronchitis

5) 1 tab PO Q12H X {duration}

Pneumocystis pneumonia

6) {Dose} PO Q6H X {duration}

Nocardia

7) 1 tab QID X 7 months

Chancroid

8) 1 tab BID X 7 days

PCP prophylaxis for HIU+ w/ CD4<200 or prev. PCP (other regimens ok)

9) 1 tab PO 3 times per week (M-W-F)

PEDIATRICS - Not indicated for children

If wt >40 kg appropriate to use adult dosing above

FIGURE 9.5. Ordering trimethoprim-sulfamethoxazole.

for the order). A given order can have multiple Sig. templates. For example, the Sig. menu for trimethoprim-sulfamethoxazole has seven options, including one for urinary tract infections and another for PCP prophylaxis (see Figure 9.5).

We use fill-in-the-blank templates for constructing the order instructions. A very simple, fill-in-the-blank template for medication may include options for the dose, the frequency, and the reason for the order (e.g., “prn pain”). The choices within the template apply to the specific order as well as the specific order context. Complex fill-in-the-blank templates can express rules for adjusting an IV drip, such as dopamine infusion based on the blood pressure response. Fill-in-the-blank templates also can be used to request the justification for an order. For example, the template for GI consults asks the user to pick one of the acceptable reasons for a GI consult, and an order for vancomycin asks the user to provide a reason that will satisfy pharmacy and therapeutics (P & T) guidelines to reduce the development of vancomycin-resistant bacteria (see Figure 9.6).

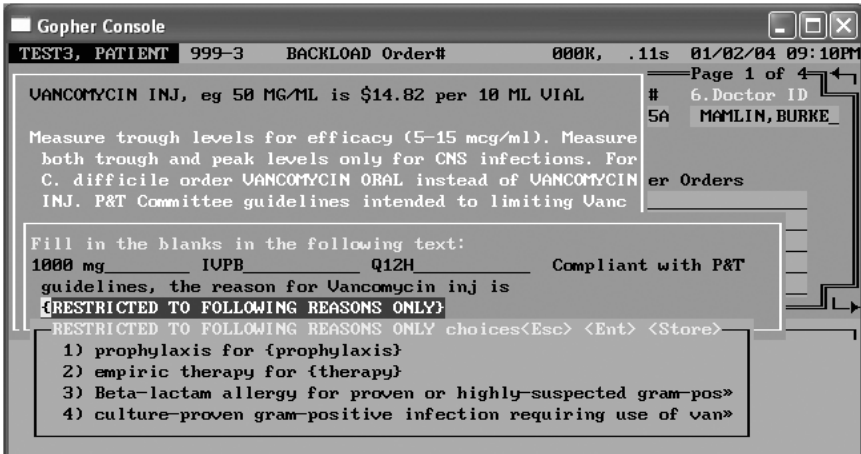


FIGURE 9.6. Ordering vancomycin.

We use a combination of executable logic, order-specific billboard text, and fill-in-the-blank templates to solve many institutional cost and operational problems, while at the same time improving physicians' workflow and reducing their paperwork.

Reference Materials and Tools

In addition to the decision support described above, we provide easy access to online references and resources, including MEDLINE, PubMed, UpToDate,[®] drug information resources, a nutritional handbook, disease management guidelines, and a clinical calculator, as well as a link to our medical library with access to hundreds of online medical journals.

Architecture of the Regenstrief Medical Record System

In this section, we offer a brief overview of the RMRS architecture, including the knowledge base, inference engines, and user interfaces used for decision support.

Knowledgebase

The knowledge within RMRS is spread throughout multiple, integrated resources. The electronic medical record (EMR) exists on a central system that is tied to all of the ancillary systems and data sources.³⁹ The PC-based Medical Gopher order-entry system runs on a local area network (LAN) connected to, yet distinct from, the central repository. The majority of our

decision-support resources are stored on the Gopher's LAN for performance purposes (see Table 9.1).

Inference Engines

There are two main “engines” running within the RMRS to deliver decision support (see Table 9.2).

TABLE 9.1. RMRS clinical decision support knowledge sources.

Knowledge	Location	Description
Specialty-specific menus	Gopher specialty table	Includes menu choices and templates that vary between medical specialties.
Problem-specific menus	Gopher dictionary	Each Gopher dictionary entry for a medical problem contains sets of comments and orders pertinent to the problem. For example, “congestive heart failure” menus include orders for diuretics, ACE inhibitors, beta blockers, daily weights, etc.
Differential diagnoses	Gopher DDX table	Symptom-based differential diagnoses data.
Medical necessity	Gopher medical necessity table	Relationships between tests, procedures and diagnoses are based on payer criteria for reimbursement.
Drug allergies	Gopher code	Drugs ordered are checked against a coded list of the patient's allergies.
Drug interactions	Gopher drug interactions table	Drug family, specific drug, and diagnosis interactions, along with descriptions of the interaction, are scanned whenever a drug order is entered into the system.
Medical and drug references	Gopher reference tables and Internet	Links to reference information are available at any time through a help key or menu option.
Consequent orders	Gopher dictionary	Consequent orders are tests or treatments that should be ordered before an order is initiated, or for monitoring after an order is initiated.
Order templates (fill-in-the-blanks)	Gopher dictionary	Templates are written in a local format and stored into the Gopher dictionary. When an order is placed, this information is pulled from the dictionary and used to generate fill-in-the-blank templates. G-CARE rules may be embedded to provide additional information and create dynamic templates.

TABLE 9.2. RMRS inference engines.

Knowledge	Location	Description
Encounter form reminder protocols	RMRS	One large file contains all reminder protocols. A batch process executes all of these protocols nightly to generate reminders for the next day's scheduled clinic visits. Protocols are written in the CARE language.
G-CARE	Gopher CARE Rule Table	Decision support logic written in an Arden-like language that allows for a complex network of expressions to generate reminders, fetch and display results or test prices, suggest orders, block contraindicated orders, and even prompt the user for data.

CARE rules running on a server generate reminders for visit notes,¹⁴ while a PC-based extension, G-CARE, drives the immediate response of suggested orders, corollary orders, blocking orders, and dynamic menus during the ordering process.²³ The notification of drug-drug interactions and patient allergies is performed by specialized codes within the CPOE system.

Communicating with the User

Users receive decision support from the RMRS in numerous ways (see Table 9.3). Generally speaking, providers receive decision support and interact with the system through two primary methods: reports (e.g., encounter forms) and the CPOE system. The RMRS prints out diagnosis lists, patient-based reminders, and dynamic data entry prompts on the encounter form for each visit. It also prints out a single-page clinical abstract summarizing recent lab results, visits, and medications.⁴⁰

In both inpatient and outpatient settings, all physicians interact with our CPOE system when writing orders or notes. All clinics have workstations in the doctor charting room, however, several have computers in the exam room as well. Reminders produced by the Medical Gopher's inference engine can be presented to an ordering provider in several formats:

1. Textual information, for example, an informational message relevant to the current order.
2. Annotated orders that can be accepted or rejected with a keystroke or mouse click.
3. A request for information, such as the patient's body weight or height.
4. An insertion of patient-specific values or a calculation result into a test display or menu item.

TABLE 9.3. The ways an RMRS provider experiences decision support.

Allergy notification
Drug-drug interactions
Suggested orders (when starting an order session for a patient)
Specialty- and problem-specific order sets
Dynamic menus (e.g., weight-based)
Diagnosis- and age-specific dosing and administration-interval suggestions
Dynamically computed dosing suggestions (e.g., adjusted for renal function)
Corollary orders (e.g., “since you ordered X, you might want to order Y”)
Blocking orders (e.g., “X is no longer formulary, consider ordering Y or Z”)
Notification of existing advance directives
Links to references/tools (library resources, patient education, clinical calculator, etc.)
One-button access to patient’s active orders and medication profile
Supersets (batteries of orders, e.g., “DIABETIC SUPPLIES MENU”)
Order-specific institutional policies
Price of orders
Date and value of most recent result(s) when ordering tests
E-mail (e.g., “a social worker informs the team that patient X was accepted to an extended care facility”) and daily messages
To-do lists and patient notes
Notification of patient encounters
Printed reminders on pocket-sized rounds reports, encounter forms, medication lists, and other reports
Alphanumeric paging for critical values
Alphanumeric paging notification for clinic patients seen in the emergency room or admitted

Timing is an important challenge in the successful use of reminders in decision support. Therefore, we have designed the CPOE system to trigger reminders in various contexts:

1. Upon selecting an orderable item: these reminders are usually designed to either discourage or redirect the provider to an alternative, and less expensive or safer order. We call these “blocking rules.”
2. Following the completion of an order: these are suggestions for actions that might be required *because* of the new order, for example, in response to an order for intravenous gentamicin, to suggest that gentamicin levels be followed.
3. After entering a problem into the patient’s problem list or declaring the reason for a visit: medical problems can trigger requests for more information or trigger problem-specific suggestions.

Any of these triggers may spawn a message via internal e-mail or over the internal digital pager to inform someone outside the clinic of an event. The principal use of this feature is to inform study coordinators when a provider has given permission for a patient to be enrolled in a clinical trial so that the researcher can approach the patient during the same visit.

Lessons from More Than a Quarter Century of Experience with Computerized Decision Support

Evidence-Based Lessons

- Computerized reminders . . .
 - . . . can change clinical behavior;¹¹
 - . . . can reduce errors and improve adherence to practice guidelines;^{3,30}
 - . . . do not necessarily provoke providers to review the associated literature;¹²
 - . . . have a strong and persistent effect on patient care;¹⁴
 - . . . can promote preventive medicine in both the outpatient¹⁴ and inpatient²⁹ settings;
 - . . . have a greater effect than delayed feedback for enhancing preventive care;¹⁶ and
 - . . . can increase discussion and completion of advance directives.³¹
- Presenting prior test results can reduce unnecessary testing.²⁴
- Offering providers predictions of abnormal results can reduce testing.²⁶
- Displaying the charges for diagnostic tests significantly reduces the number and cost of tests ordered, especially for patients with scheduled visits.²⁵ This effect does not persist if charges are no longer displayed.
- Reminders for flu shots can generate better patient outcomes.²⁰
- Requiring physicians to respond to computer-generated reminders improves their compliance with preventive care protocols;¹⁸ however, promoting preventive care through computerized reminders presents further challenges in the inpatient setting.²⁸
- CPOE and CDSS can significantly reduce patient charges and hospital costs.²⁷
- CPOE-based CDSS can be attained with little or no time added to the patient care process.⁴¹

Experience-Based Lessons

- Start with the assumption “the user is always right” because computer systems often lack fine details and reminder rules cannot anticipate every situation.
 - Users should be able to override nearly every decision. When we first created reminders for mammograms, we found that users were dismissing them. Why? Because the computer system was unaware of a prior mastectomy, a dying patient, or a recently obtained mammogram from another institution. Simply appending these conditions as selectable options after the reminder both acknowledged the system’s limitations and regained our users’ confidence.

- Workflow is paramount!
 - Keep it simple.
 - Workflow is one of the most critical aspects of delivering excellent, efficient patient care. Decision support often introduces new steps (whether it is a new piece of paper to be reviewed or an alert within CPOE that must be navigated). Implementers of decision support must be cognizant of the impact on workflow.
 - Whenever possible, favor provider-oriented workflow.
 - Avoid punishing the user with additional obstacles when simple rewording or changing a default value will do. The same result often can be achieved in either a user-friendly or not-so-friendly manner. For example, if providers are disregarding decision support that suggests a more effective test or treatment, first consider where, when, and how the message is being delivered (e.g., could it be conveyed more concisely or at a more appropriate position in workflow?) before introducing extra steps (e.g., forcing the user to acknowledge the message with an extra key press).
 - When possible, allow for free text. For example, we generally allow order instructions to be selected from the menu or typed in as free text. While free text may limit the computer's understanding of the data, judicious use of free text can make the system more "user-friendly."
- Response-times should be sub-second, i.e., "blazingly fast."
 - Providers do not want to wait for a computer during a busy day. Decision support that involves significant processing should either be simplified or moved to a batch process that can run asynchronously with the patient's visit.
- Don't overwhelm the users.
 - Too many reminders or too many choices are worse than none.
 - Keep messages and text short and focused.
- Never underestimate the power of user feedback—seek it out!
 - Early in the development of the Medical Gopher, we invented the pizza meeting for gaining user feedback. In these weekly pizza meetings, we traded pizza for house staff and student feedback on the system. This feedback was critical in both forming a user-friendly system and addressing system problems early.
 - Listening is often 90% of the solution; however, the ability to respond rapidly with improvements or fixes will cover the last 10%.
- Up-time is critical.
 - To effectively incorporate decision support into workflow, providers must be able to depend on the system. Erratic behavior (e.g., unpredictable reminders) will not be tolerated and users will quickly dismiss the system.

References

1. Prokosch HU, McDonald CJ. The effect of computer reminders on the quality of care and resource use. In: Prokosch HU, Dudeck J, eds. Hospital information systems. Amsterdam: Elsevier Science Publishers B.V.; 1995:221–239.
2. McDonald CJ, Bhargava B, Jeris DW. A clinical information system (CIS) for ambulatory care. Proceedings of the American Federation of Information Processing Societies National Computer Conference at Anaheim, California; May 19–22, 1975.
3. McDonald CJ. Protocol-based computer reminders, the quality of care and the non-perfectability of man. *N Engl J Med* 1976;295:1351–1355.
4. Cole RB, Balmer JP, Wilson TS. Surveillance of elderly hospital patients for pulmonary tuberculosis. *Br Med J* 1974;1:104–106.
5. Craven RB, Wenzel RP, Atuk N. Minimizing tuberculosis risk to hospital personnel and students exposed to unsuspected disease. *Ann Intern Med* 1975; 82:628–632.
6. Jacobs S, Greenberg HB. Diagnosis and treatment of 20 tuberculous patients who entered a community hospital. *Am Rev Respir Dis* 1972;105:528–531.
7. Greenblatt DJ, Koch-Weser J. Adverse reactions to spironolactone. A report from the Boston Collaborative Drug Surveillance Program. *JAMA* 1973;225: 40–43.
8. Shapiro S, Slone D, Lewis GP, Jick H. Fatal drug reactions among medical inpatients. *JAMA* 1971;216:467–472.
9. Caranasos GJ, Stewart RB, Cluff LE. Drug-induced illness leading to hospitalization. *JAMA* 1974;228:713–717.
10. Kelley CR, Mamlin JJ. Ambulatory medical care quality. Determination by diagnostic outcome. *JAMA* 1974;227:1155–1157.
11. McDonald CJ. Use of a computer to detect and respond to clinical events: its effect on clinician behavior. *Ann Intern Med* 1976;84:162–167.
12. McDonald CJ, Wilson GA, McCabe GP Jr. Physician response to computer reminders. *JAMA* 1980;244:1579–1581.
13. McDonald CJ. Action-oriented decisions in ambulatory medicine. Chicago: Yearbook Medical Publishers; 1981.
14. McDonald CJ, Hui SL, Smith DM, Tierney WM, Cohen SJ, Weinberger M, et al. Reminders to physicians from an introspective computer medical record. A two-year randomized trial. *Ann Intern Med* 1984;100:130–138.
15. Zeger SL, Liang KY. Longitudinal data analysis for discrete and continuous outcomes. *Biometrics* 1986;42:121–130.
16. Tierney WM, Hui SL, McDonald CJ. Delayed feedback of physician performance versus immediate reminders to perform preventive care. Effects on physician compliance. *Medical Care* 1986;24(8):659–666.
17. Berwick DM. Continuous improvement as an ideal in health care. *N Engl J Med* 1989;320(1):53–56.
18. Litzelman DK, Dittus RS, Miller ME, Tierney WM. Requiring physicians to respond to computerized reminders improves their compliance with preventive care protocols. *J Gen Intern Med* 1993;8:311–317.
19. Litzelman DK, Tierney WM. Physicians' reasons for failing to comply with computerized preventive care guidelines. *J Gen Intern Med* 1996;11:497–499.

20. McDonald CJ, Hui SL, Tierney WM. Effects of computer reminders for influenza vaccination on morbidity during influenza epidemics. *MD Comput* 1992;9:304–312.
21. Shapiro S. Periodic screening for breast cancer: the HIP Randomized Controlled Trial Health Insurance Plan. *J Natl Cancer Inst Monogr* 1997;22:27–30.
22. McDonald CJ, Tierney WM. The Medical Gopher—a microcomputer system to help find, organize and decide about patient data. *West J Med* 1986;145(6): 823–829.
23. Overhage JM, Mamlin B, Warvel J, Tierney W, McDonald CJ. A tool for provider interaction during patient care: G-CARE. *Proc Annu Symp Comput Appl Med Care* 1995:178–182.
24. Tierney WM, McDonald CJ, Martin DK, Rogers MP. Computerized display of past test results. Effect on outpatient testing. *Ann Intern Med* 1987;107:569–574.
25. Tierney WM, Miller ME, McDonald CJ. The effect on test ordering of informing physicians of the charges for outpatient diagnostic tests. *N Engl J Med* 1990;322:1499–1504.
26. Tierney WM, McDonald CJ, Hui SL, Martin DK. Computer predictions of abnormal test results. Effects on outpatient testing. *JAMA* 1988;259:1194–1198.
27. Tierney WM, Miller ME, Overhage JM, McDonald CJ. Physician inpatient order writing on microcomputer workstations. Effects on resource utilization. *JAMA* 1993;269:379–383.
28. Overhage JM, Tierney WM, McDonald CJ. Computer reminders to implement preventive care guidelines for hospitalized patients. *Arch Intern Med* 1996;156: 1551–1556.
29. Dexter PR, Perkins S, Overhage JM, Maharry K, Kohler RB, McDonald CJ. A computerized reminder system to increase the use of preventive care for hospitalized patients. *N Engl J Med* 2001;345:965–970.
30. Overhage JM, Tierney WM, Zhou XH, McDonald CJ. A randomized trial of “corollary orders” to prevent errors of omission. *J Am Med Inform Assoc* 1997;4:364–375.
31. Dexter PR, Wolinsky FD, Gramelspacher GP, et al. Effectiveness of computer-generated reminders for increasing discussions about advance directives and completion of advance directive forms. A randomized, controlled trial. *Ann Intern Med* 1998;128:102–110.
32. Tierney WM, Overhage JM, Takesue BY, et al. Computerizing guidelines to improve care and patient outcomes: the example of heart failure. *J Am Med Inform Assoc* 1995;2:316–322.
33. McDonald CJ, Overhage JM, Tierney WM, Abernathy GR, Dexter PR. The promise of computerized feedback systems for diabetes care. *Ann Intern Med* 1996;124(1 Pt 2):170–174.
34. Tierney WM, Overhage JM, McDonald CJ. Computerizing guidelines: factors for success. *Proc AMIA Annu Fall Symp* 1996:459–462.
35. Tierney WM, Overhage JM, Murray MD, et al. Effects of computerized guidelines for managing heart disease in primary care: A randomized, controlled trial. *J Gen Intern Med* 2003;18:967–976.
36. Tierney WM, Overhage JM, Murray MD, et al. Computer-based outpatient care suggestions for reactive airways disease. *J Gen Intern Med* 2002;17:213–214.
37. Murray MD, Harris LE, Overhage JM, et al. Failure of computerized treatment suggestions to improve the health outcomes of outpatients with uncomplicated

- hypertension: results of a randomized controlled trial. *Pharmacotherapy* 2004; 24:324–337.
38. Kroenke K, West SL, Swindle R, et al. Similar effectiveness of paroxetine, fluoxetine, and sertraline in primary care: a randomized trial. *JAMA* 2001;286: 2947–2955.
 39. McDonald CJ, Tierney WM, Overhage JM, Martin DK, Wilson GA. The Regenstrief Medical Record System: 20 years of experience in hospitals, clinics, and neighborhood health centers. *MD Comput* 1992;9:206–217.
 40. McDonald CJ, Overhage JM, Tierney WM, et al. The Regenstrief Medical Record System: a quarter century experience. *Int J Med Inf* 1999;54:225–253.
 41. Overhage JM, Perkins S, Tierney WM, McDonald CJ. Controlled trial of direct physician order entry: effects on physicians' time utilization in ambulatory primary care internal medicine practices. *J Am Med Inform Assoc* 2001;8: 361–371.

10

Decision Support During Inpatient Care Provider Order Entry: The Vanderbilt Experience*

RANDOLPH A. MILLER, L. RUSSELL WAITMAN, SUTIN CHEN, and S. TRENT ROSENBLOOM

Introduction

The need for decision support systems in medicine has been understood for 2,500 years. Hippocrates noted “Life is short, the art long, opportunity fleeting, experience treacherous, *judgment difficult*.” (*Aphorisms*, sec. I, ca. 460–400 BC). While the basis for clinical decision support has been recognized throughout the ages, careful studies in the recent medical literature document those needs specifically.^{1–14} Early pioneers, such as McDonald, Tierney, and their colleagues at the Regenstrief Medical Institute^{15–25} (see Chapter 9); Warner, Gardner and their colleagues at LDS Hospital^{26–28} (see Chapter 8); and many other groups have confirmed, through controlled studies, the initial report of Shakespeare in 1597: “If to do were as easy as to know what were good to do, chapels had been churches, and poor men’s cottages princes’ palaces. . . . I can easier teach twenty what were good to be done than to be one of the twenty to follow my own teaching” (*The Merchant of Venice*, Act I, Scene ii). Busy clinicians have so many diverse tasks to perform that they are constantly distracted from being able to accomplish what is known to them as good medical practice. “Men are men; the best sometimes forget” (Shakespeare, *Othello*, 1605; Act II, Scene iii). Reminding systems and other forms of clinical decision support have been shown to be effective in overcoming such lapses of memory in a number of clinical situations.^{15–40} However, the success of systems intended to be used by busy practitioners is not guaranteed. A significant number of clinical

* This chapter is an adaptation for this book of an article reprinted from the Journal of Biomedical Informatics Vol. 38, Miller RA, Waitman LR, Chen S, Rosenbloom ST. The anatomy of decision support during in patient care provider order entry (CPOE): Empirical observations from a decade of CPOE experience at Vanderbilt, 469–485, 2005, with permission from Elsevier, used with permission.

informatics systems (not all documented in the literature) implemented with good intentions have been met with anger and resentment.^{41–44} Providing decision-support capabilities in a timely and convenient manner can add value to otherwise lackluster or marginal systems, and improve quality of care and reduce costs.^{15–40}

This chapter addresses the following questions: (1) What steps or stages in physician or other care provider order entry (CPOE) represent appropriate breakpoints (both computationally and with respect to end-user workflows) at which one can introduce clinical decision support? (2) What categories of decision support are relevant during CPOE sessions? and (3) What methods for workflow interruption should one consider in order to implement decision-support interventions based on end-user tolerance and clinical urgency?

The authors use the Vanderbilt WizOrder CPOE system as the primary context for discussing decision support interventions. Through longstanding partnerships with clinician end-users, Vanderbilt Biomedical Informatics faculty members, fellows, and staff developed a CPOE system (WizOrder), implemented it on the wards of an academic teaching hospital, and evolved it in response to ongoing feedback.^{45–55} The approach to decision support described in this chapter was derived through generalization from experience. While the authors draw heavily on their Vanderbilt experience, the above questions and their answers are generic enough that others may find value from the descriptions provided herein.

The authors describe the precommercial version of WizOrder at Vanderbilt. The WizOrder CPOE system was developed by Vanderbilt University faculty within the School of Medicine beginning in 1994. The product was not commercialized until May–June of 2001. At that time, Vanderbilt University entered into a marketing agreement with McKesson to commercially sell and distribute the WizOrder Care Provider Order Entry System, after rewriting and recasting it as Horizon Expert Orders in McKesson's product line. The authors have referred to the system throughout the manuscript by its name at Vanderbilt, WizOrder. All descriptions are of system components developed at Vanderbilt University and not by the commercial vendor.

Basic Care Provider Order Entry System Functionality

Order entry within most CPOE systems parallels manual paper chart-based order creation. Manual ordering involves: (1) finding the patient's chart; (2) finding the topmost blank order page; (3) handwriting new orders as a block; (4) signing the orders to make them legal; (5) after setting a flag indicating presence of new orders, placing the chart where clerical unit staff expect to find charts with new orders; and (6) informing unit staff (patient's nurse, others) when life-critical or extremely urgent orders have been written. For the corresponding electronic CPOE processes,

the user: (1) authenticates with user name and password; (2) invokes the CPOE application and selects a patient for order entry; (3) enters and modifies orders, using an electronic scratchpad (buffer) that holds orders but does not deliver them to ancillary departments (e.g., lab or pharmacy) for immediate action; (4) indicates when he or she is ready to finalize the set of orders on the scratchpad to send them out for processing; and (5) reviews and edits orders on the scratchpad before they are dispatched to be carried out. With electronic CPOE, person-to-person manual communication of life-critical, or otherwise very urgent, orders remains essential for patient safety.

The panes of Figure 10.1 represent the Vanderbilt approach to implementing key components of an order entry interface. Vanderbilt end users strongly recommended that the CPOE system interface should have geographical consistency—meaning that the same types of clinical information are always displayed in the same areas of the screen, and that the number of pop-up windows and pull-down menus that could obscure display of clinically important information were limited. WizOrder’s left-sided window displays currently active (or expired in the previous 24

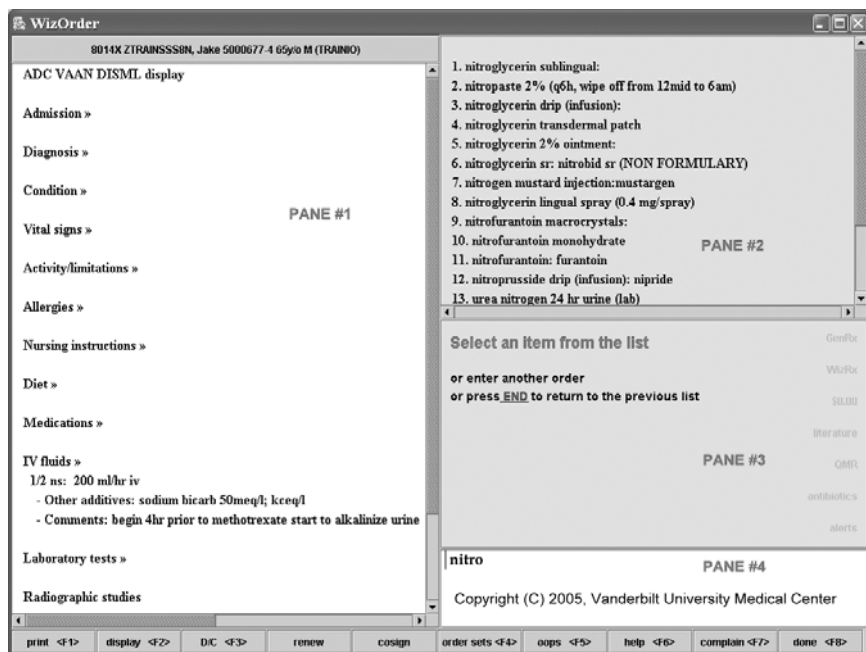


FIGURE 10.1. WizOrder primary user interface screen panes: PANE #1, current and recent orders display; PANE #2, selectable “pick list” display; PANE #3, in-context instructions; PANE #4, user input text entry area. User had previously typed “nitro” into completer in PANE #4; PANE #2 shows results.

hours) orders for the current CPOE patient (PANE #1). The upper right window presents context-dependent pick lists of options available for order creation or modification (PANE #2). The middle right window represents a context-sensitive help window that instructs the user on available next actions (PANE #3). The bottom right window contains a text input region (PANE #4).

Creating Orders

A key CPOE system design consideration involves how clinicians specify what they want to order. Many systems⁵⁶⁻⁵⁹ utilize a hierarchical organization of orders, illustrated by the following example (bold font indicates hypothetical selection made at each level):

... Orderable Pick List Level 1: Pharmacy, **Laboratory**, Radiology, Dietary, Nursing [orders], ...
 ... Orderable Pick List Level 2: **Hematology Tests**, Serum Chemistry Tests, Urinalysis, ...
 ... Orderable Pick List Level 3: **Complete blood count (CBC)**, platelet count, blood Rh type, ...

Many systems also have a completer function that allows the clinician-user to type shorthand word fragments derived from the desired order name (or its synonyms). The completer then searches for potentially matching terms from the orderables dictionary, and provides the user with a pick list of order names from which the user can select. For example, typing “nitro” into an orders completer (Figure 10.1, PANE #4) would return a pick list (PANE #2) of orderable items’ names, with “nitroglycerin sublingual” at or near the top of the list, and lesser/partial/wordier matches (e.g., nitrogen mustard, urea nitrogen blood) farther down the list. Vanderbilt users typically specify new orders using the completer function, and only rarely use WizOrder’s hierarchies for order entry—usually when they do not know the specific name for the item they want to order.

After selecting an order name, users must specify (enter) an individual order’s components (e.g., dose, route, frequency, etc, for a medication order). Many CPOE systems formally define orderables and their components using a data dictionary with structured templates that specify necessary and optional fields required to fully create an individual order. Figure 10.2 illustrates WizOrder sequential prompts for building an order for sublingual nitroglycerin (based on stored templates), and Figure 10.3 indicates how the order, once fully specified for WizOrder, transfers to the left-sided active orders area (PANE #1). Another mechanism for generating new orders (used often, but less than half the time at Vanderbilt) is order sets—groupings of diagnosis or procedure-related selectable orders.⁶⁰ If the user selects an order set name from a completer pick list or from the WizOrder

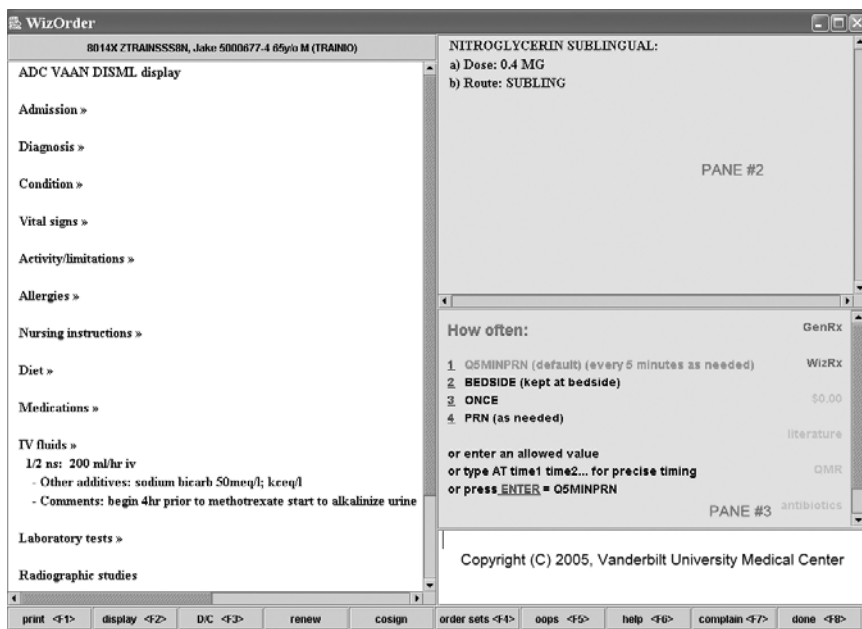


FIGURE 10.2. Frequency prompts (medication-specific) for “nitroglycerin sublingual” orderable, after dose already specified by similar process.

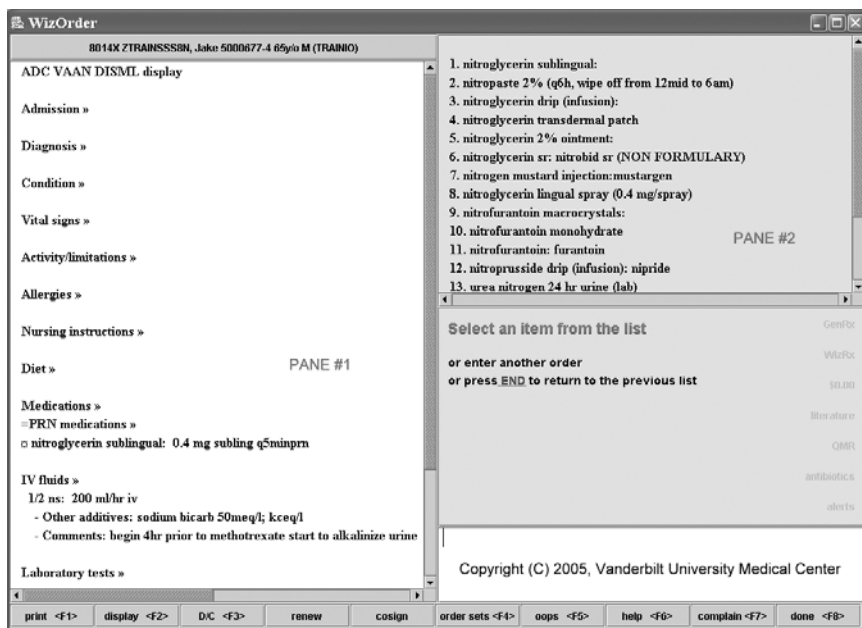


FIGURE 10.3. Order for “nitroglycerin” moves to left window (PANE #1) once fully completed.

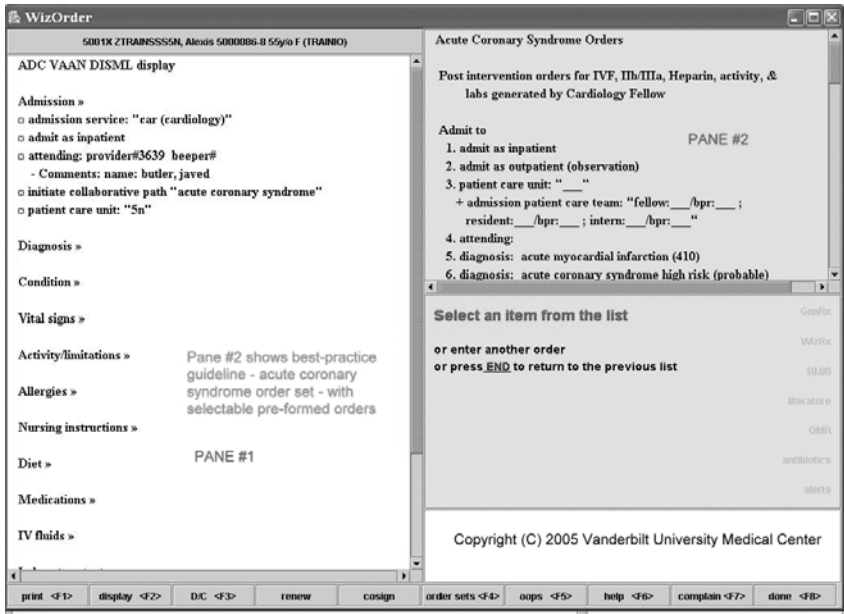


FIGURE 10.4. First six orders in the acute coronary syndrome order set.

order set hierarchy, the order set's component orders are retrieved and displayed as selectable items in the upper right pick list window (Figure 10.4, PANE #2).

Displaying Active Orders

Most CPOE user interfaces manage the display of currently active orders. In complex patient cases, the number of active orders may exceed 100. Therefore, simply listing all such orders in a display panel (sorted alphabetically by order name) will not be helpful to clinicians unfamiliar with the patient's case, since locating an arbitrarily named specific order within a long list is difficult. Early in the development of WizOrder, end users requested that a display of active orders follow a grouping based on the ADC VAAN DISML acronym (familiar to physicians)—**A**dmission, **D**iagnosis, **C**ondition, **V**ital signs, **A**ctivity, **A**llergies, and so on (Figure 10.1, PANE #1). Most CPOE systems use similar methods to segment the active orders display into clinically useful buckets. Many CPOE systems facilitate electronic rearrangements of the active orders display to accommodate different users' workflows (e.g., nurses, attending physicians). Vanderbilt's specialized intensive care units and the emergency department, for

example, required location-specific specialized views of active orders. As WizOrder displays active orders, it also displays recently expired orders (within the past 24 hours) with a special symbol in the left margin to indicate that those orders that have expired; a different left-margin symbol indicates orders soon to expire.

Modifying and Finalizing Orders

Figure 10.5 illustrates the result of a mouse click on an order in the left WizOrder pane. WizOrder displays a series of options listing what the user can do to the order at that point (modify, discontinue, renew, etc.) After the WizOrder user is finished entering orders for a session, clicking a designated button on the CPOE screen transfers the user to a final accept screen (see Figure 10.6). This screen gives users a last chance to verify (or to change) their orders from the current ordering session. Once final-accepted, the orders are sent to the appropriate ancillary systems for action and committed to a relational database for archiving. Similar features are available in most CPOE systems.

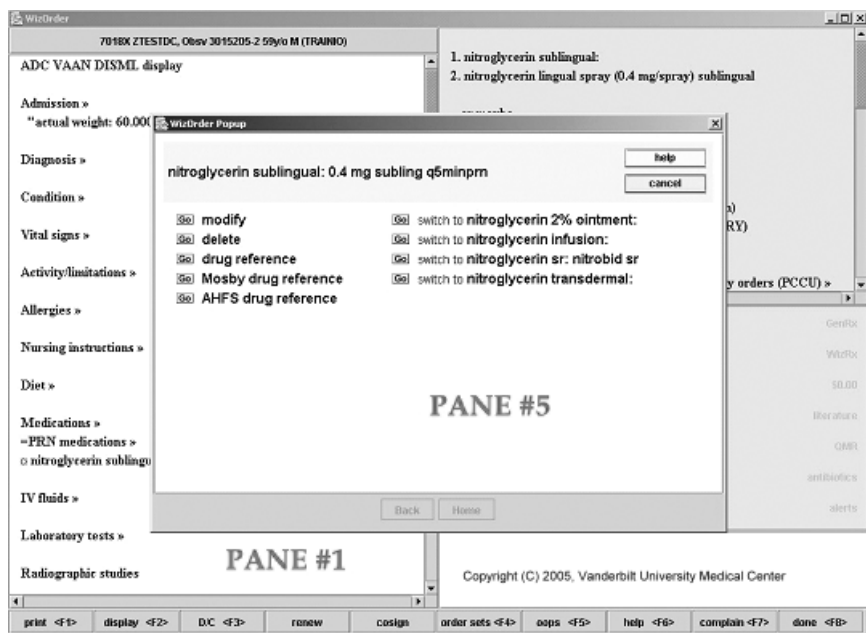


FIGURE 10.5. “Pop-up” options (PANE #5) after selecting nitroglycerin order from PANE #1.

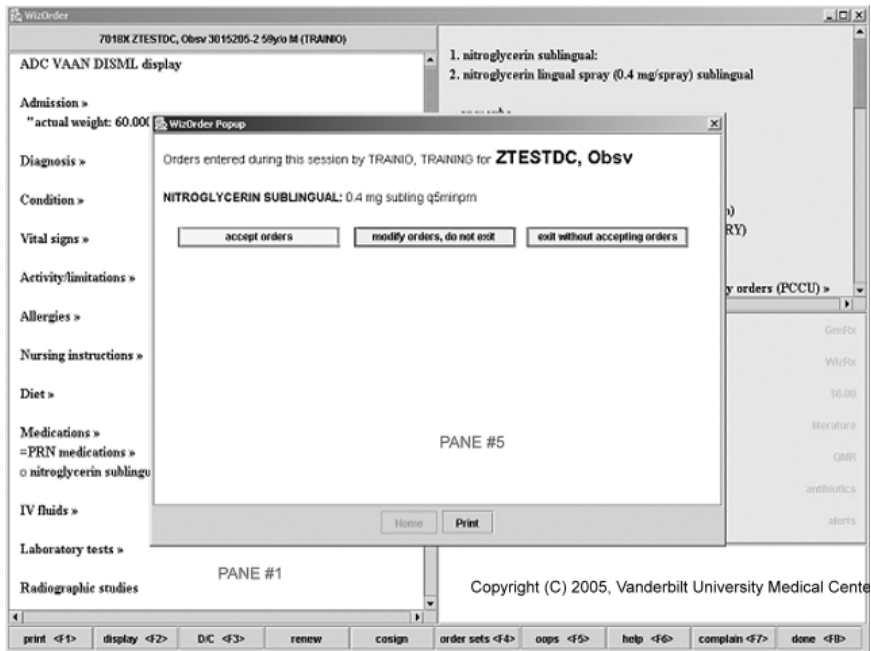


FIGURE 10.6. Final accept screen (PANE #5) allows user to verify orders at end of ordering session.

Displaying Information and Providing Complex Decision Support

A final WizOrder component consists of an intermittently displayed, pop-up window that contains an internal HTML browser (labeled “PANE #5” in various figures). The WizOrder program uses this capability to display static Web documents with educational content or dynamically generated CPOE-related pages that provide complex, patient-specific decision support capabilities.^{49,50}

Philosophy Underlying Decision Support During Care Provider Order Entry

Use of a CPOE system during patient care provides a unique opportunity to interject decision support features that improve clinical workflows, provide focused relevant educational materials, and influence how care is delivered to patients. It is somewhat of an art to be able to provide clinical decision support that is well accepted and used widely. Key considerations in the approach to providing decision support include: what content to provide; when to intervene in the clinical workflow process; and how to

intervene, in terms of both degree of disruption of workflows and mechanism of interruption. These considerations are addressed later in this chapter.

The nature of the clinical application domain determines what types of decision-support content to provide. It is not appropriate to allow a clinician to spend 1–2 minutes constructing an intricate medication order, only to later inform the clinician that the medication is contra-indicated due to a previously documented patient allergy. Allergy warnings should take place at the time the clinician first indicates the name of a new medication to be prescribed. Conversely, delivering a warning to a clinician to order a partial thromboplastin time (PTT) monitoring test, immediately as the clinician completes a heparin order, may cause both exasperation and a lost sense of autonomy when that is exactly what the clinician intends to order next. Checking whether a PTT monitoring test has been ordered at the end of an order entry session, during which intravenous heparin therapy was initiated, may be more appropriate, since the user is done entering orders at that stage. Oppenheim et al. observed that permitting the physician to enter an order with feedback provided only at the conclusion of order construction, and then only if the order is possibly incorrect, serves dual purposes.⁶¹ First, delayed warnings make clinicians first commit to a preferred course of action, thus discouraging reliance on CPOE systems to make clinical decisions for the users. Second, delayed warnings give the clinician user the opportunity to correct problems they detect spontaneously, whereas early warnings may impart negative reinforcement by underscoring clinicians' errors.⁶¹

In the authors' experience, busy clinical users value CPOE system responsiveness and intuitiveness. A key aspect of responsiveness involves creating orders at an appropriate clinical level (both for users' levels of training and for their knowledge of their patients). The physicians and nurses entering orders into a CPOE system typically have a different mindset than individuals who will carry out the orders in ancillary areas (e.g., pharmacy, radiology, and dietary departments). Problems in creating CPOE system orderable item names can occur when the technical terms used in ancillary departments are carried forward as the orderable items vocabulary for clinicians. So while radiology technicians might think in terms of "chest X-ray 2 views" and "knee X-ray 3 views", clinicians are more comfortable ordering "chest X-ray PA and lateral" and "knee X-ray AP, lateral and oblique." Similarly, if the CPOE system asks the physician ordering a chest X-ray how the patient should be transported to the radiology department, the physician is unlikely to give an optimal response because physicians are rarely involved in determining a patient's transport. Thus, CPOE systems should not ask clinicians to perform tasks that fall outside of their job responsibilities, or about which they have little knowledge. Structuring orderable items with the clinician in mind helps to overcome major barriers to adoption and can prevent errors.

Intelligent system interfaces can dramatically decrease the burden of ancillary departments in dealing with CPOE system generated orders. For example, pharmacists use the pharmacy system to fill and dispense the clinical orderables specified within the CPOE system; if a high-level order is issued by the physician, it may require more work on the pharmacy system side of the interface to specify all components of an order correctly. Once the physician specifies a medication order at a clinical level, an intelligent interface within the CPOE system can evaluate both the pharmacy's formulary and the floor stock inventory on the patient's unit, and then automatically determine the correct dispensable within the pharmacy system. Currently, the intelligent pharmacy interface within WizOrder guesses the correct pharmacy-level dispensable item over 90% of the time. This allows the pharmacist to devote more time to evaluating each order's clinical validity, safety, and efficacy.

As a frequently used clinician data entry tool, an institution's CPOE system becomes a target for administrators and researchers wishing to capture additional data at the point of care. It is important to avoid overburdening clinicians with requests that interrupt their workflows, and, when extra information is required, the system should only ask clinicians for information about which they are the definitive source. For example, at Vanderbilt, upon patient admission, the attending physician of record was originally input into the admission, discharge, transfer (ADT) system by an admitting clerk. However, the admitting clerks were not always informed of the specifics of physician group coverage schedules, and often they did not know the correct name to enter. The problem was addressed by finding a more definitive data source—the admitting house staff team, who must discuss each admission with the attending physician—and having them enter the name into the CPOE system. Conversely, if one wants to record whether a patient received aspirin in the emergency department just prior to admission, asking an intern who is entering discharge orders for the patient several days later (and who did not admit the patient) could be viewed as a nuisance, and cause lower-than-optimal data quality.

While some decision support functions not directly related to order entry can be delivered during an order entry session, they will not be discussed in this chapter: for example, a laboratory system that generates alerts whenever abnormal patient results occur might notify clinicians responsible for the patient's care either by paging them or via e-mail or an asynchronous pop-up alarm that occurs when the clinician is currently logged into the CPOE application.⁶² Such alerts originate outside of the CPOE session context. Many CPOE systems, including WizOrder, display permanent taskbars, an array of useful links, continuously during the application session;^{45,59,63–65} however, such taskbars rarely provide context-specific decision support of the sort described here. Instead, they allow the user to access common CPOE functions. For instance, the BICS (Brigham Integrated

Computer System, in Boston) toolbar allows the clinician to quickly view orders and search for patients, among other functions.^{64,66}

Roles for Decision Support Within Care Provider Order Entry: Categories of Interventions

1. Creating Legible, Complete, Correct, Rapidly Actionable Orders

A CPOE system can avert problems associated with handwritten order creation,⁶⁷ for example, illegibility, incompleteness, and incorrectness. Improved legibility not only reduces errors, but also saves staff time because nurses, pharmacists, and medical technicians spend time and energy as they decipher the meaning of ambiguous handwritten orders and then make phone calls to clarify what was meant. Complete orders contain all the necessary parameters to make an order actionable (order name, start date and time, duration, frequency, etc.). Correct orders have parameter values that meet requirements for safe, prudent patient care (e.g., drug doses are appropriate for the patient's age, weight, and renal function). Most CPOE system interfaces ensure completeness and promote correctness of orders.⁶⁷⁻⁶⁹

2. Providing Patient-Specific Clinical Decision Support

An important CPOE system capability is generation of decision support recommendations customized to individual patients' specific conditions. A CPOE system can provide a safety net through behind-the-scenes reconciliation of patient-specific information (laboratory results, age, allergies, existing medications from the clinical data repository⁷⁰) with stored best practice rules. For example, most CPOE systems screen patient orders against dosing rules and drug interaction references to reduce medication prescribing errors.^{53,66,71-74} A CPOE system can also facilitate clinical care improvement by promoting use of evidence-based clinical practice guidelines^{58,75,76} through end-user order generation via diagnosis or procedure-specific order sets^{56,59,65,70,76} or via computer-based advisors,^{58,64,73,77,78} as detailed below.

3. Optimizing Clinical Care (Improved Workflow, More Cost-Effective and Regulatory-Compliant)

Often, complex software systems functionally become high-level programming languages for their end users. Once clinicians regularly use a CPOE system, they begin to make suggestions about modifying it to make their

work easier and more effective. For example, to improve workflows, several surgical services at Vanderbilt encouraged WizOrder developers to create registry orders. The orders place patients on a registry that allows clinicians to track, via a census, diagnoses and procedures performed on registry patients (e.g., neurosurgery service). At the same time, registries enabled efficient transfer of appropriate information to the billing office, relieving physicians of that responsibility. Early CPOE users at Vanderbilt requested printed rounding reports to facilitate patient care during work rounds and attending (teaching) rounds. The rounding reports concisely summarize, on the front and back of an 8.5 × 11 inch piece of paper, both the patient's active orders and all laboratory results reported in the prior 72 hours with highlight markers next to significant (e.g., abnormal) results. After several years, the institution's administration began to view the CPOE system as a tool to implement quality of care, cost containment, and compliance initiatives.⁵²⁻⁵⁴ Institution-wide CPOE interventions can: discourage the ordering of inappropriate, recurring tests;^{20,52,79} advise against costly tests or require further justification before allowing them to proceed;^{22,55,80} display formulary information;^{55,57} and help the ordering clinician to enter requisite third party payer compliance codes (e.g., ICD-9 or CPT) for diagnostic tests. Clinicians are not always familiar with compliance rules, and they tend to write reasons for tests based on suspected diagnoses (e.g., "rule out MI" for an electrocardiogram, or "possible pneumonia" for a chest X-ray) rather than indications for testing approved by third party payers. Orders that require specific reasons for compliance can be made to trigger the WizOrder internal Web browser to display and capture order-specific compliance-related reasons for testing. This can increase the rate of third party payer reimbursements for those tests due to more accurate, complete capture of compliant reasons.

Clinical decision support features within CPOE systems can also promote implementation and enforcement of local hospital policies. The Regenstrief Medical Record System (RMRS) (see Chapter 9), successfully used computer reminders to increase discussion about, and completion of, advanced directives (end-of-life, "do not resuscitate" related orders).⁸¹ Previous studies had indicated that too few patients completed advance directives.⁸² In Boston, the BICS was modified in order to prevent the appearance of vancomycin-resistant microorganisms by requiring clinicians ordering vancomycin to enter a reason for using the antibiotic.⁸³

The challenge for CPOE system developers is to honor the care improvement goals while keeping the system responsive and intuitive. Developers must strike a proper balance between clinical improvements versus cost containment. At times, both goals may be achieved in a single intervention—judiciously ordering fewer tests does not mandate a lower quality of care.⁵² However, care improvement interventions may themselves have unintended consequences that require continuous monitoring and feedback for optimal results.⁵⁴

4. Providing Just-in-Time, Focused Education Relevant to Patient Care

Most CPOE systems provide relevant educational prompts and, in addition, links to more detailed educational information resources. Educational prompts can be introduced as in-line summaries that appear while prescribing a medication. Figure 10.7 shows in the upper right WizOrder panel in-line suggestions for vancomycin dosing adjustments in neonates with meningitis or with renal impairment. The CPOE Web browser content can also provide effective educational information, for example, presenting a summary of disease-specific national guidelines, links to educational monographs, or a summary of indications and contra-indications for a specific therapy. Educational links can assist clinician users to perform complex ordering, such as for total parenteral nutrition (TPN) in a neonatal intensive care unit. The design of a CPOE system user interface can significantly influence the rate at which users follow educational links and read the related materials. Simply having an option for decision support may not be

WizOrder
4NEN ZTESTSS, 7 Do 3013/20-2 1mo F (TRABNO)

Neonatal ICU display

Admission »
"actual weight: 4.000kgs/8lb13." »Jul 1 09:18...
"gestational age: 41 weeks, 3 days"
"peds dosing weight: 4.300kgs;" »Jul 1 09:17...

Diagnosis »

Condition »

Vital signs »

Activity/limitations »

Allergies » PANE #1

Nursing instructions »

Diet »

Medications »

Fluid intake: 0.0 ml/kg/day (all calculations use actual weight)
TPN: 0.0 ml/kg/day
Lipids: 0.0 ml/kg/day
Drips: 0.0 ml/kg/day
IV fluids: 0.0 ml/kg/day
Feeds: 0.0 ml/kg/day

IV fluids »

Respiratory orders »

VANCOMYCIN INJECTION:
Weight=4.000 kgs on Tue Jul 1 09:18
Peds dosing weight=4.300 kgs on Tue Jul 1 09:17
No Creat or BUN result available.
Non-CSF infections: 15 mg/kg/dose IV Q8H

Meningitis: 20 mg/kg/dose IV Q8H

Renal impairment:
Give one 15mg/kg dose and redose based on levels

PANE #2

Dose:	GenRx
a 15 MG/KG (default)	WizRx
b 20 MG/KG	\$0.00
or enter another value (in MG)	
or press ENTER = 15 MG/KG	
	literature
	QMR
	antibiotics
	alerts

FIGURE 10.7. In-line recommendations for dosing vancomycin in NICU include: (a) PANE #2, suggested doses for regular use, for meningitis, and for renal impairment; (b) PANE #1, passive display of weight, dosing weight, and gestational age; and (c) PANE #2, display of renal function test results (not available for training patient in this example).

sufficient to command users' attention, and stronger cues, such as different visual displays as to the relevance of the information, may be needed.⁵⁵

Critical Points at Which to Implement Decision Support Within Care Provider Order Entry

Each stage of use of a CPOE system permits a focused repertoire of decision support interventions, both in terms of user community affected, patients affected, and appropriateness of the intervention for the task the end user is performing. For example, as the CPOE system is launched from a clinical workstation desktop, system-wide messages can appear, but patient-specific advice cannot (since, typically, a patient has not yet been selected). Below, we discuss the type of decision support that is appropriate and feasible for each stage.

1. Stage of Care Provider Order Entry Session Initiation

Upon initial launching of the CPOE application, the identity of the clinician user is known, but not the identity of the patient. At this stage, users may be advised of new CPOE system features on a one-time-only basis. To avoid annoying users, such interventions should be used sparingly; for example, for features of general interest to all users such as a new method of entering a specific group of commonly used orders that replaces the previous method of doing so. Once the alert is displayed, the system removes the current user from the list of users who still need to see that message again. At launch, the CPOE system can also inform users of information related to their personal use of the system, such as the number of orders (and number of patients) requiring their countersignature, and provide a link to facilitate completing the task.

2. Stage of Selecting Care Provider Order Entry Patient from Hospital Ward Census

After CPOE system launch, users typically select an individual patient for order entry. A number of alerts can occur at the stage of displaying the census of available patients for CPOE. Similar to the BICS system in Boston (and other CPOE systems), WizOrder provides, via the patient census screen, an inpatient, unit-wide view of the status of recently issued orders (see Figure 10.8). A map view of the given hospital ward shows all beds and uses color coding to indicate which beds have new unacknowledged, urgent (i.e., stat) orders and which have unacknowledged routine orders. A care provider wishing to enter new orders (or acknowledge recent orders) can click on a bed on the display screen to initiate an order entry session for that particular patient. An alternative to the map view of a hospital unit census is a list view that lists patients

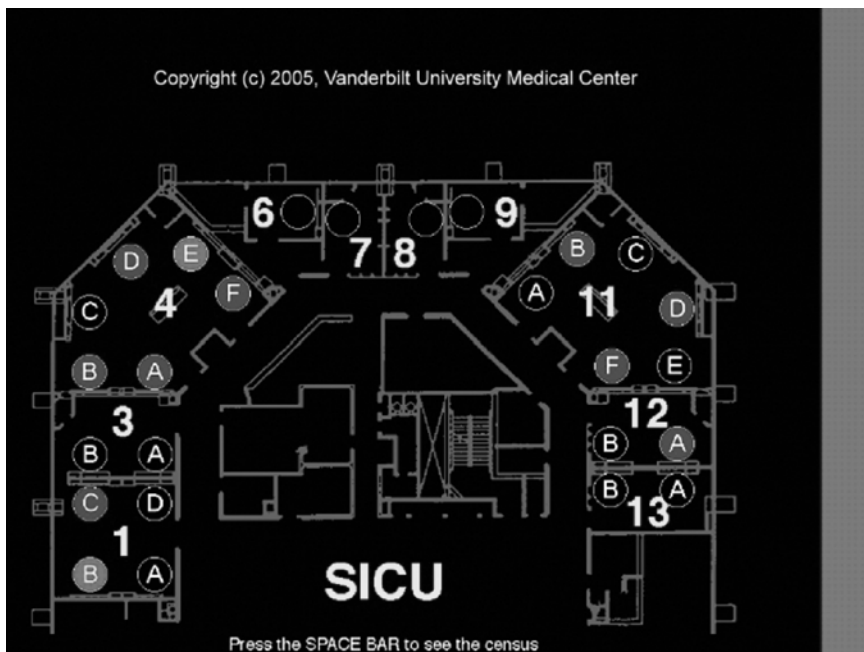


FIGURE 10.8. CPOE “map” view of hospital ward. Map indicates beds (circles) with different shading to indicate new, urgent “stat” orders or those with new “routine” orders; right border shading indicates highest priority of new orders not yet acknowledged (across all beds) by nursing staff.

on the unit, and that can be sorted by patient name or by ascending bed number. In WizOrder, icons located beside patients’ names in the list view provide useful information (Figure 10.9). Using a similar list census screen, the BICS system presents a renewal reminder next to the patient’s name when a medication order for a given patient nears expiration.⁸⁴

3. Stage of Individual Patient Session Initiation

Once the order entry session becomes specific to a selected patient, several new types of decision-support related events can occur. In WizOrder, once the patient is identified, the system retrieves all relevant past (active and inactive) orders for the patient, and previously stored patient-specific information such as weight, height, coded allergies, and active protocols (with related date of protocol initiation information). As the user waits for the initial patient-specific CPOE screen to appear, WizOrder queries the patient data repository to obtain the patient’s recent laboratory results for common important tests, in order to assist with subsequent CPOE decision-support recommendations.

Nursing station: 7N			TRAINIO census	Stations	
V	ZORDERS, A	7025B 70 G	I	ZTRAINMCK, Demochild	10N
V	ZORDERS, C	7024B 7I G	I	ZTRAINSS8N, Danielle	10S
	ZTESTMOR, Hipaa	7028X 7I G	O	ZTESTMINE, Lf	10SD
	ZTRAINED, Chris	7030X 70 GE	O	ZTESTPHM, Wilson	11NM
V	ZTRAINPCIS, Donot Alter	7008B IX S			3N/C
V	ZTRAINPCIS, Donot Alter	7009B IA G			4CN
V	ZTRAINPCIS, Donot Alter	7020X EA N			4EN
V	ZTRAINPCIS, Donot Alter	7029X OX G			4EN2
V	ZTRAINPCIS, Donot Alter	7031X OT G			4EST
	ZTRAINSS7N, Ashton	7006X IA N			4NI
	ZTRAINSS7N, Bethany	7003X 7I N			4NIA
	ZTRAINSS7N, Caleb	7004X 7I D			4NIC
	ZTRAINSS7N, Daphne	7005X 7I N			4NPL
	ZTRAINSS7N, Ethan	7010X IA U			5N
	ZTRAINSS7N, Florence	7007X 7I N			5PCU
	ZTRAINSS7N, Greg	7016X 7I S			5S
	ZTRAINSS7N, Holly	7011X 7I M			6A
S	ZTRAINSS7N, Ian	7012X 7I D			6N
	ZTRAINSS7N, Jada	7013X 7I N			6S
	ZTRAINSS7N, Kevin	7014X 7I N			7A
	ZTRAINSS7N, Lori	7015X 7I N			7B
	ZTRAINSS7N, Morgan	7027X 7I N			7C
	ZTRAINSS7N, Nora	7017X 7I G			7N
	ZTRAINSS7N, Owen	7018X IA N			7SHI
	ZTRAINSS7N, Paige	7019X 7I N			8A
	ZTRAINSS7N, Quinton	7021X DI S			8B
	ZTRAINSS7N, Rosie	7022X 7I S			8C
	ZTRAINSS7N, Sebastian	7002X 7I U			8N
	ZTRAINSS7N, TheLma	7023X 7I N			8S
	ZTRAINSS7N, Ursula	7026X 7I N			9NEM
					9S
					ADMT
					CCL

FIGURE 10.9. “Patient list” view of CPOE ward census. Several graphical “icon” alerts (left margin next to patient name) provide useful information regarding ward census at a glance. The inverted triangles provide duplicate last name warnings; “S” indicates patients on whom medical students have entered orders that must be reviewed by a licensed medical doctor to become “activated;” and pumpkins indicate patients who have been bedded as outpatients long enough that conversion to inpatient status (or discharge to home) should be considered.

Ability to recover from an interrupted CPOE session without loss of work (time and effort) is critical to busy clinicians’ acceptance of such systems. Lost sessions can occur due to system bugs (such as a disk becoming unexpectedly full), environmental factors (such as network outages or power failures), and user factors (such as abandoning a workstation during a medical emergency, with a subsequent session timeout). Figure 10.10 shows the alert that occurs upon initiation of a patient-specific CPOE session for a patient with a previously interrupted session. The user is then given the option to play back and recover the orders from the previously interrupted session.

Among the many other types of alerts that can occur at the stage of initiating a patient-specific CPOE session are: presentation of a summary of past alerts and warnings related to the patient’s orders—e.g., allergies and drug interactions; notification of medications about to expire; display of the names of active protocols for the patient (e.g., “Deep Venous Thrombosis prophylaxis protocol”); and promotion, via reminders, of new protocols for which the patient is eligible. Figure 10.11 illustrates an admission wizard that indicates to the user, for the ward on which the patient is bedded,

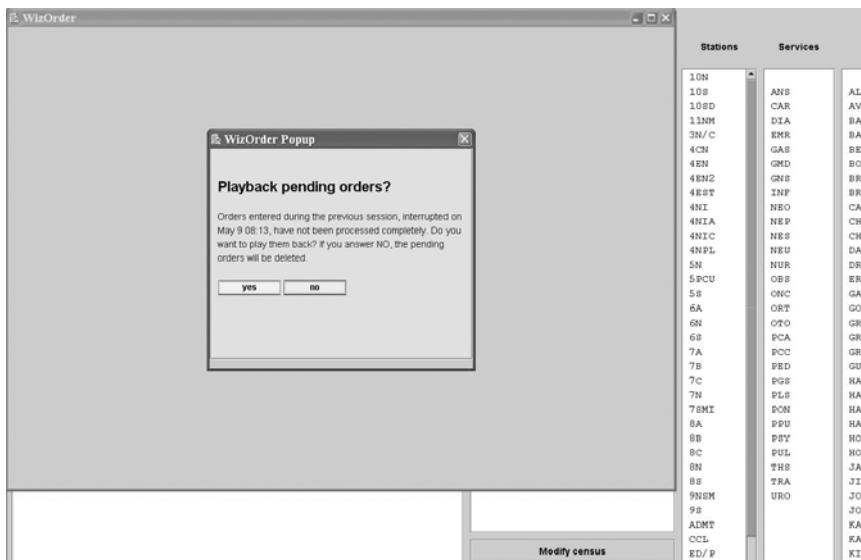


FIGURE 10.10. Interrupted/incomplete previous WizOrder CPOE session warning. Allows user to recover from previously interrupted ordering session.

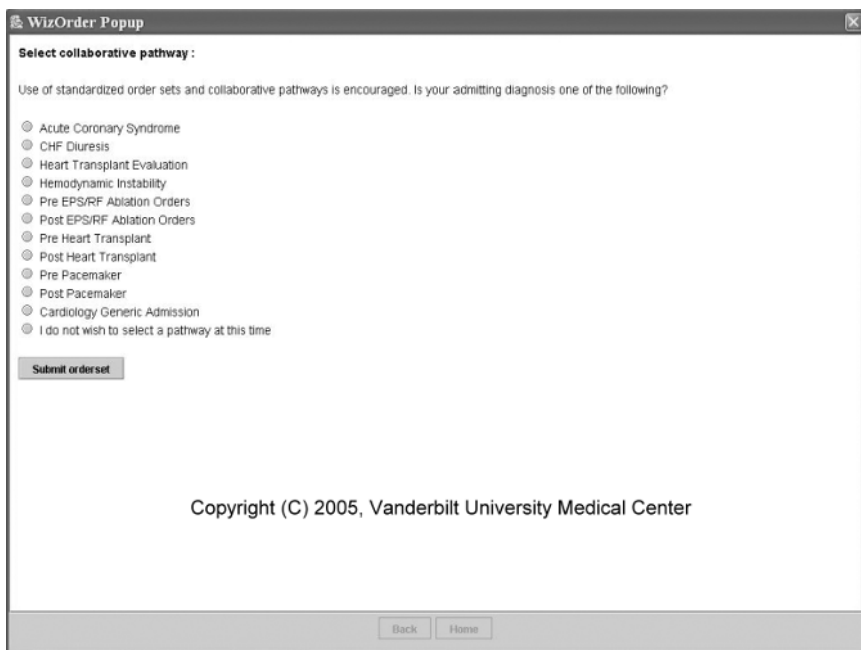


FIGURE 10.11. Admission Wizard prompts user to select evidence-based protocol for patient when relevant to case.

the commonly used, evidence-based best-of-care order sets that are available within the system, and it encourages the user to select one for use on the patient, if applicable. The structure of such an order set, once selected, is shown in Figure 10.4 in the upper right window (PANE #2).

4. Stage of Individual (Single) Order Selection

Upon selecting a specific CPOE orderable item, and before the details of the order are specified by the user, certain decision support checks may be appropriate. In order to not waste the user's time, once a drug name is identified as being the next order, and before the user specifies the details of the drug order (dose, route, frequency, etc), CPOE-based allergy and drug interaction checking should issue any relevant warnings. Figure 10.12 illustrates a WizOrder drug-drug interaction warning after entry of the new medication name.

Individual order selection can also trigger protocol-based interventions such as recommending drug substitutions (suggesting a less expensive or more effective medication than the one originally selected). Similarly, single order selection can initiate computer-based advisors related to the specific order (Figure 10.13A and 10.13B). A similar mechanism that redirects physician workflow occurs in the BloodLink-Guideline system for test ordering.⁵⁸ Many CPOE systems offer the capability to link order sets to

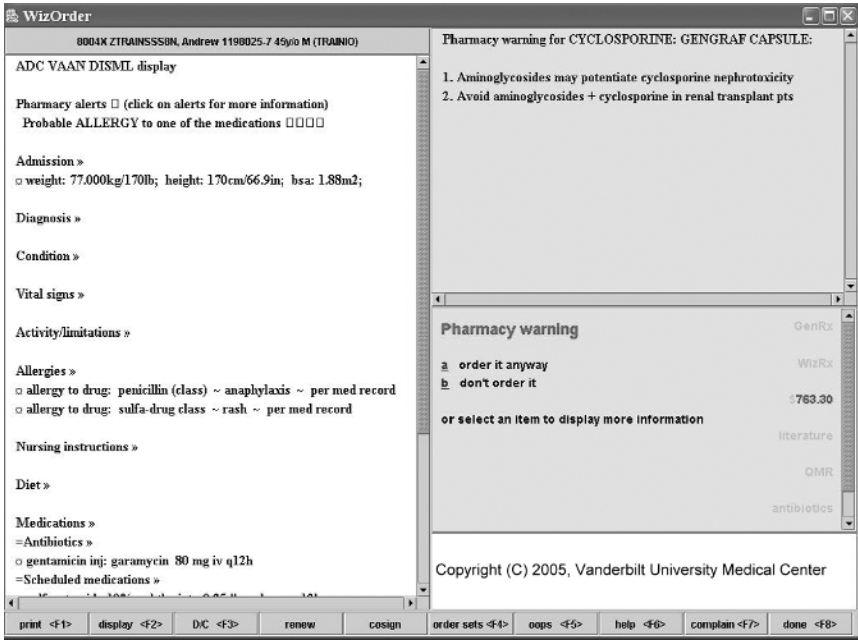
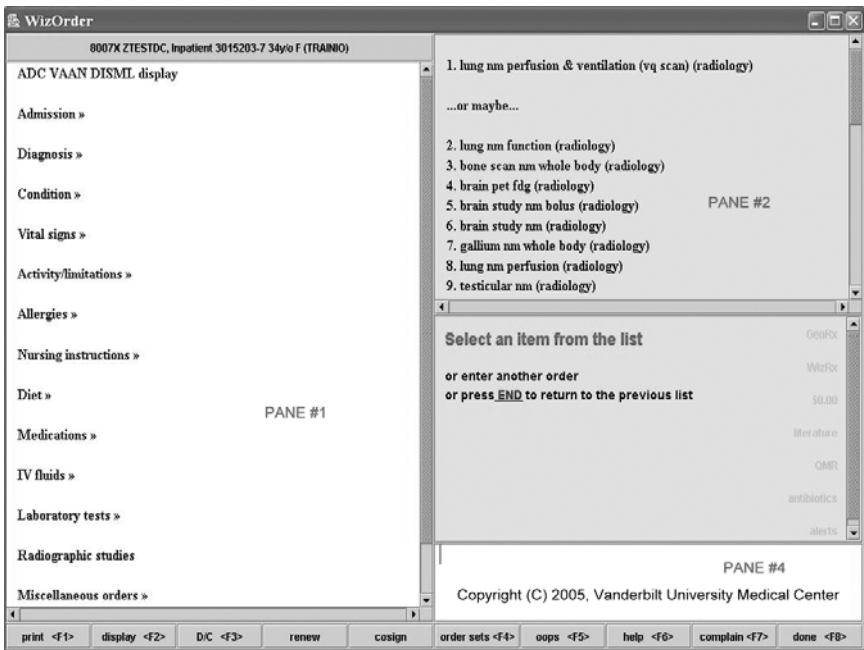
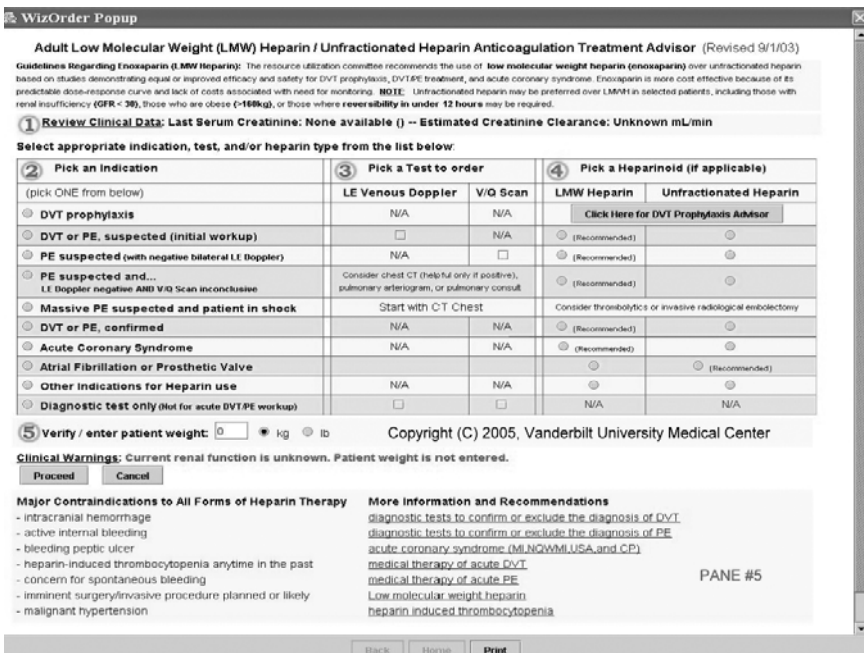


FIGURE 10.12. Drug-drug interaction warning after entry of new medication name.



A



B

FIGURE 10.13. (A) Clinician-user initially attempted to order “VQ scan” of lung for pulmonary embolism, and WizOrder completer maps to official name of test (item 1 in PANE #2), which user then selects by typing choice in PANE #4. (B) Selecting lung scan order from A launches anticoagulation adviser in WizOrder, helps clinician select appropriate diagnostic workup, and therapy for suspected or confirmed deep venous thrombosis (DVT) or pulmonary embolism (as well as DVT prophylaxis and therapy for other disorders such as acute coronary syndrome).

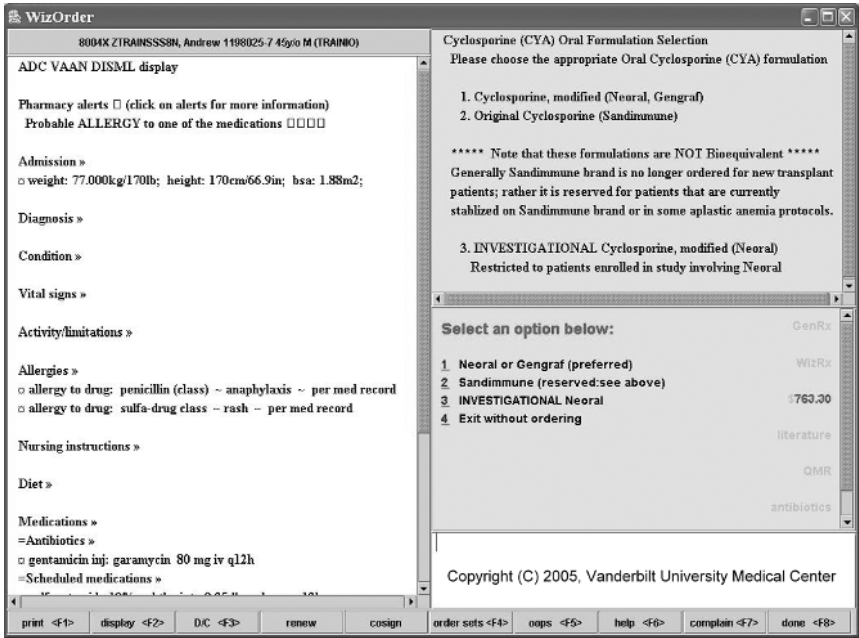


FIGURE 10.14. “In-line”, patient-specific, interactive advice for clinician while attempting to prescribe cyclosporine for patient; developed by experts in the pharmacy to guide clinician to best choice.

individual selectable orders (i.e., to transfer the user to an order set when an individual order is selected).^{56,59,65,70,76} Order sets are described in detail below.

5. Stage of Individual (Single) Order Construction

Once the order name has been selected, the CPOE system assists the user in completing required steps for order construction (see Figure 10.14 for an example of instructions during cyclosporine ordering). WizOrder guides medication order construction by highlighting recommended drug doses and drug frequencies and by presenting alerts for potentially incorrect decisions. This is similar to what is described in the literature for the BICS system in Boston.^{66,73} Many CPOE systems also provide computer-based advisors to enforce compliance with established, evidence-based guidelines.^{58,77} As described in Chapter 8, the antibiotic advisor system at LDS Hospital in Salt Lake City recommends therapy options for critically ill patients based on patient vital signs and serology, microbiology, pathology, and radiology results.⁷⁷

Based on their research, Bates et al. observed that clinicians generally take the path of least resistance.^{73,85,86} Providing effective decision support involves not only alerting the provider about a potential error, but pro-

viding a correct alternative option as well. For instance, in the BICS system, if meperidine hydrochloride is prescribed for a patient whose creatinine clearance (a measure of renal function), is significantly diminished, an alert notifies the user that the drug might possibly promote seizures in this patient, and suggests a substitute medication.⁸⁴

6. Stage of Individual Order Completion

Once an individual order's components have been fully specified (and any allergy or other alerts that might have prevented order construction have been dealt with), a number of decision-support functions related to the order as a whole become appropriate. Upon completed order construction, many CPOE systems suggest corollary orders—follow-up tasks clinically indicated after certain orders.^{73,84,87,88} For example, after ordering gentamicin, an antibiotic, it is often appropriate to order serum drug levels. Figure 10.15 illustrates this capability in WizOrder. As discussed in Chapter 9, the RMRS system presents corollary orders for many drug-drug monitoring test pairs (e.g., warfarin prescriptions and related INR/prothrombin time tests) and for drug-drug side effect pairs (e.g., prescription of class II narcotics and orders for stool softeners to treat/prevent the constipation caused

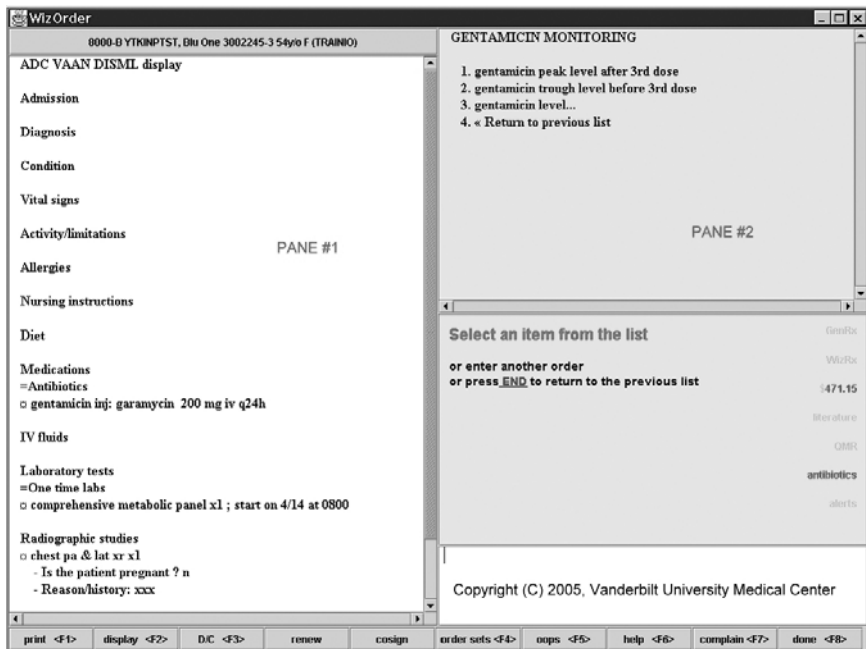


FIGURE 10.15. After completing gentamicin order (seen in PANE #1), system offers selectable gentamicin monitoring orders (in PANE #2) as “one click away” for convenience (suggesting best practice, but not requiring it).

by narcotics).⁸⁷ Another example is offering clinicians the opportunity to order heparin (to prevent DVT) after a completed order for bed rest (which predisposes to DVT).⁸⁴ Research has shown more effective ordering and improved outcomes as a result of such systems.⁸⁹

7. Stage of Ordering Session Completion

Once the user has specified all individual new (or modified) orders and wishes to finalize the ordering session, various decision-support related exit checks are appropriate. As noted above, recurring reminders to do what the clinician user already intended to do are not well tolerated. Instead of using corollary orders to prompt PTT and INR monitoring after orders for heparin and warfarin, respectively, WizOrder waits until the ordering session is complete. At that point, it becomes fair game to issue warnings if appropriate monitoring tests have not been issued. Conversely, if during a given ordering session, a clinician discontinues either the heparin infusion or the PTT monitoring tests but not the other item of the pair, it is appropriate to use an exit check that warns the clinician that parallel actions to discontinue both are usually needed. Figures 10.16 and 10.17 illustrate the

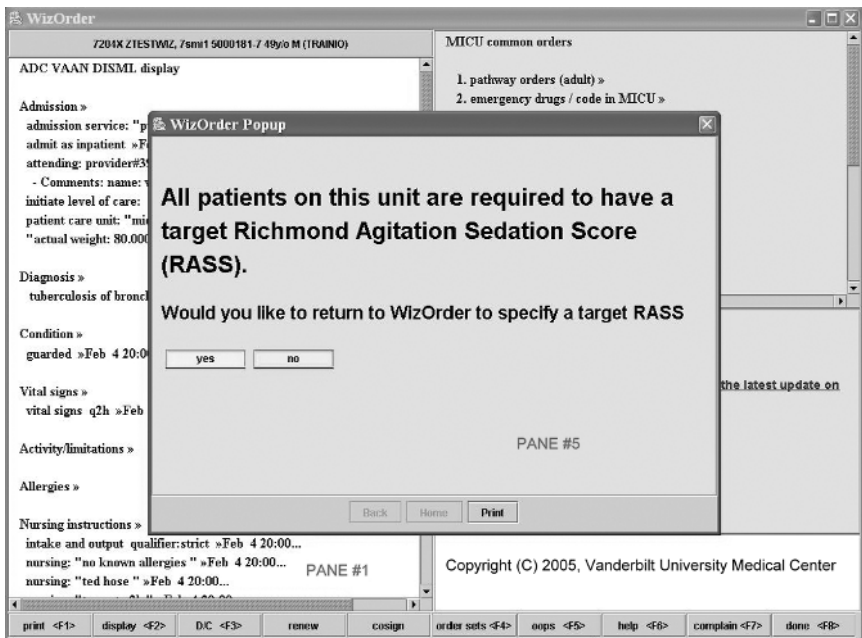


FIGURE 10.16. WizOrder “exit check.” On completing admission orders on an ICU patient, if the clinician-user has not specified a target RASS (Richmond Agitation Sedation Scale) score, the system uses a pop-up alert to remind the clinician that it is ICU policy to do so.

Please select the target Richmond Agitation Sedation Scale (RASS) value for ZTESTWIZ, 7smi1

Target RASS Value	RASS Description
<input type="radio"/> 0 Alert and Calm	
<input type="radio"/> - 1 Drowsy	Not fully alert, but has sustained awakening to voice (eye opening/contact > 10 seconds)
<input type="radio"/> - 2 Light Sedation	Briefly awakens to voice (eye opening/contact < 10 seconds)
<input type="radio"/> - 3 Moderate Sedation	Movement or eye opening to voice (but no eye contact)
<input type="radio"/> - 4 Deep Sedation	No response to voice, but movement or eye opening to physical stimulation
<input type="radio"/> - 5 Unarousable	No response to voice or physical stimulation

Procedure for RASS Assessment

- Observe Patient
 - Is patient alert, restless, or agitated? (**Score 0 to + 4**)
- If not alert, state patient's name and say to open eyes and look at speaker.
 - Does patient awaken - sustained eye opening and contact? (**Score - 1**)
 - Does patient awaken with eye opening and contact, but not sustained? (**Score - 2**)
 - Does patient awaken (eye opening and contact), but has eye opening or movement in response to voice (**Score - 3**)
 - Physically stimulate patient by shaking shoulder and/or rubbing sternum
 - No response to voice, but response (movement) to physical stimulation (**Score - 4**)
 - No response to voice or physical stimulation (**Score - 5**)

05/04/05 09:58 PANE #5
 Copyright (C) 2005, Vanderbilt University Medical Center

FIGURE 10.17. User (from Figure 10.16) requests assistance in specifying RASS score; web-based advisor assists user with data collection and score calculation.

two-part WizOrder exit check for ordering (or updating) the Richmond Agitation and Sedation Scale (RASS) target score whenever pain medications or sedatives are ordered for a patient in an ICU.

Care Provider Order Entry Intervention Approaches— from Subtle to Intrusive

While the interfaces of successful CPOE systems are rarely seamless, users adapt to their styles of workflow after training and repeated use. Once acclimated to the CPOE system workflows, users do not appreciate interruptions that deter them from the previously noted path of least resistance.⁸⁶ Determining whether, how, and when to disrupt clinician workflows to provide appropriate decision support is critical to end-user acceptance of both the decision support and the CPOE system overall. Below, we describe a number of approaches to introducing decision support, from nondisruptive to very disruptive, and give examples of where each may be appropriate.

1. *Incidental Display of Relevant Information*

Presentation of additional viewable text on a portion of the usual application screen allows the user direct access to relevant information with

minimal interruption to workflow. Because no user input (e.g., acknowledgment of the information) is required, and no additional information is available (e.g., the user cannot click on or select the displayed information to learn more), the clinician is free to read or to ignore the displayed information. WizOrder displays the most recent results of serum electrolyte tests during ordering of intravenous fluid therapy. WizOrder also displays relevant dosing information for prescribing medications, for example, on pediatric units, the patient's actual weight, dosing weight, and pharmacy-recommended dosing guidelines (see Figure 10.7). Information relating to costs may be displayed as well.

2. Incidental Display of Linked Education Opportunities

A CPOE system may have order-related educational information that is too voluminous to include in the usual order entry screen. Under such circumstances, the CPOE system can present links for users to select (click on) that lead to a separate screen/window providing the relevant textual information. Examples might include links to relevant drug guidelines and formulary information.⁵⁹ The Vanderbilt Patient Care Provider Order Entry with Integrated Tactical Support study,⁵⁵ provided links to pharmacotherapy-related information (illustrated by the “GenRx” and “WizRx” links on the right margin of PANE #3, Figure 10.2), and reference material for diagnosis in internal medicine. Figure 10.18 provides an example, in PANE #5, of displaying an evidence-based summary of what is known about a specific drug interaction (selected by the user from the drug interaction warnings list of Figure 10.18, PANE #2). In other systems, as clinicians review recommended drug doses for patients with renal impairment, they can display the data used to calculate creatinine clearance using a keyboard shortcut link.⁶⁴

3. Interactive Sequential Advice for User-Directed Clinical Activity

By presenting stepwise instructions in context, CPOE systems help users to carry out discrete tasks. Figure 10.2 presents the default minimum type of advice that WizOrder provides for order construction; Figure 10.14 provides a more complex example whereby the user is sequentially prompted, through questions and answers, to order the most appropriate form of cyclosporine for the patient. Another system, the BloodLink-Guideline system⁵⁸ directs blood test ordering decisions by first having the clinician select the appropriate guideline, then presenting a menu of related indications, and, finally, presenting a menu of relevant tests for a selected indication.

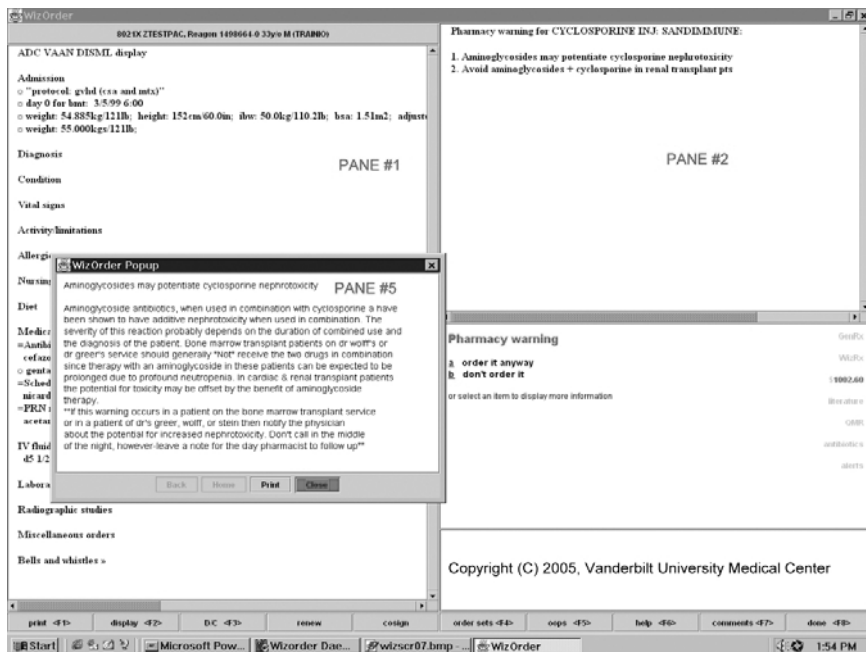


FIGURE 10.18. Clinician prescribed cyclosporine while a currently active order for gentamicin was in place. Following a drug interaction alert (PANE #2), user clicks on item 1 to request evidence basis for what is known about the drug interaction (displayed in pop-up window, PANE #5).

4. Recallable Best Practice Guidelines with Actionable Pre-Formed Pick List Selections

Order sets are pick lists containing constituent individual pre-specified full orders, often representing standardized protocols. Figure 10.4 illustrates a portion of the WizOrder order set for acute coronary syndrome. Order sets are often presented in hierarchies for easy access, organized by clinical department,^{40,59} by organ system, or by clinical diagnosis, condition, or procedure.^{57,76,90} While picking orders from order sets may be viewed as disruptive to the usual workflow of creating individual orders, in many CPOE systems appropriate use of order sets can increase users' time-efficiency and promote completeness and correctness of orders.^{58,60,91}

5. Pop-Up Alerts that Interrupt Workflow and Require a Response for the User to Continue

Pop-up alerts can present clinically important information (in a separate user interface window) that must be acknowledged by the user before

resuming previous CPOE activity. Use of such interventions is typically viewed by users as disruptive, and should be reserved for only the most severe clinical indications. Pop-up fatigue can occur when too many alerts of this type disrupt clinical workflows.⁹² In WizOrder and other systems, pop-up windows alert physicians when excessive chemotherapy doses are ordered.^{48,93} Figure 10.19 illustrates how a WizOrder user is notified that the most recent laboratory test ordered will be sent out to a reference laboratory for completion, and provides advice on how to optimize ordering with respect to institutional policies regarding reimbursement for testing. This mechanism is used to display hospital-approved drug substitution regimens. Figure 10.20 shows a WizOrder drug substitution pop-up (implemented as an advisor, Method 6, below). Figure 10.16 shows how the RASS exit check was implemented as a pop-up alert in WizOrder. Figure 10.12

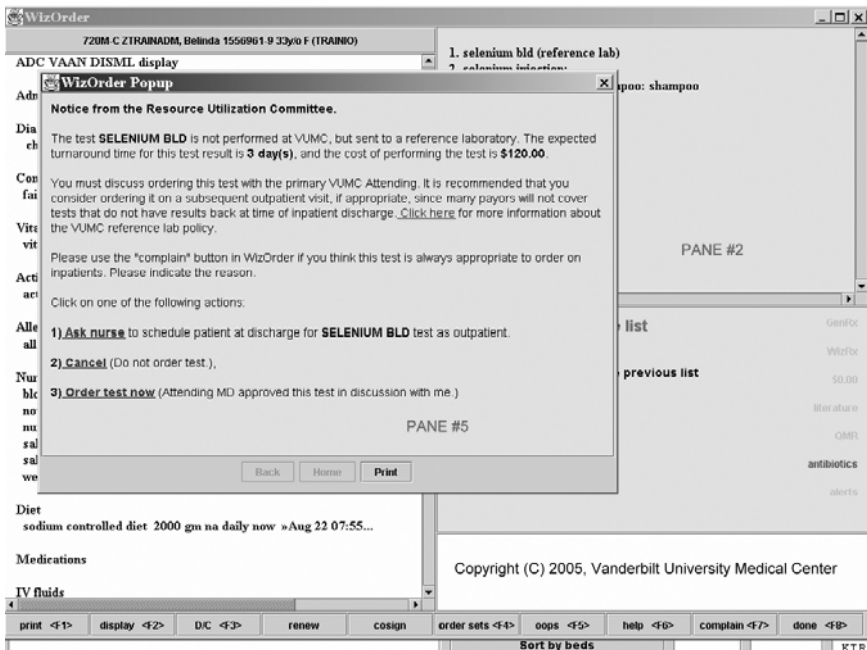


FIGURE 10.19. Clinician user begins to order “selenium blood” level (PANE #2), prompting a pop-up warning (PANE #5) that stops workflow and demands attention. The pop-up explains that the test is sent to a reference laboratory and takes three days to perform. User is notified that reimbursement may be compromised if patient is discharged before result is known. Pop-up provides instructions for alternative ordering mechanisms (that can be selected directly from pop-up) if clinician believes that obtaining the result of the order is not urgent/emergent for the current patient.

WizOrder Pop-up

The VUMC Antibiotic Subcommittee recommends Cefepime (Maxipime ®) over Ceftazidime (Fortaz ®) for most indications where an anti-pseudomonal cephalosporin is needed.*

Cefepime 1000 mg q12h = Ceftazidime 1000 mg q8h

*Exception for neonates and selected pediatric patients. Safety and effectiveness of Cefepime in pediatric patients below the ages of 2 months have not been established.

Compared to ceftazidime, Cefepime has the following advantages:

- Similar coverage against *Pseudomonas*, improved coverage against *Enterobacter* species
- Enhanced stability against inducible/derepressed chromosomal beta-lactamases
- Better activity against Gram-positive pathogens, including *Staphylococci*, *S. viridans*, *Streptococcus*
- Q12 hour dosing except for empiric therapy for febrile neutropenia

[View Cefepime Fact Sheet](#) [Go to Pediatric Recommendations](#) [Go to Renal Dosing Recommendations](#)

Adults (Age > 18 years)

Dose	Example of infection being treated
<input type="radio"/> 500 mg IV q12h	Uncomplicated urinary tract infection
<input checked="" type="radio"/> 1000 mg IV q12h	Nosocomial pneumonia in ICU patient PANE #5
<input type="radio"/> 1000 mg IV q8h	Empiric coverage of febrile neutropenic patient
<input type="radio"/> 2000 mg IV q8h	The FDA approved a dose of 2 gm IV q8h for febrile neutropenic patients and this is preferred over the 1 gm IV q8h dose if cefepime is given as monotherapy for this indication. The 1 gm IV q8h dose has been used in the Bone Marrow Units and is appropriate for febrile neutropenic patients receiving other antibiotics with activity against Gram-negative aerobic pathogens such as aminoglycosides or quinolones. Documented infection with <i>Pseudomonas aeruginosa</i> should be treated with the higher (2 gm IV q8h) dose.

Other

Intramuscular order IM Cefepime (with Lidocaine)

Non-standard Dose order non-standard dose of Cefepime

[Order Cefepime](#) [Start Over](#) [Order Ceftazidime](#)

Click the CLOSE button to return to WizOrder without ordering cefepime

Copyright (C) 2005, Vanderbilt University Medical Center

[Back](#) [Home](#) [Close](#)

Sort by beds

FIGURE 10.20. User ordered an antibiotic for which the Pharmaceuticals and Therapeutics (P&T) Committee had recommended a substitution. A variant pop-up, this educational advisor guides the clinician through ordering an alternative antibiotic. Links to “package inserts” (via buttons) detail how to prescribe the recommended drug under various circumstances. A physician who knows little about the recommended drug could learn enough to prescribe it appropriately.

illustrates how WizOrder uses the pop-up method to present a drug interaction alert.

6. Complex, Computer-Based Protocols that Interact with the User to Make Patient-Specific Calculations and Recommendations

The most complex form of decision support is an interactive advisor that integrates patient-specific information (laboratory results, active orders, weight, allergies, etc.) with complex guidelines or protocols, and presents calculated/derived information to the user for decision making, typically involving a two-way dialogue between the application and the user. Complex advisors may combine educational advice, calculators for patient-specific dosing, and other functionality in one screen. The Antibiotic

WizOrder Pop-up PANE #5

TPN fluid requirement must be a least 20 ml/kg/day.

1 TPN fluid requirement: ml/kg/day (not including lipids)

Cycle TPN over hours

Central Line TPN Order Sheet

2 <Review Current Lab Trends>

Patient: ZTESTSS, 7 Do (female) TPN Calculation Weight: 3.8 kg

3 Amino Acids as Trophamine grams/kg/day
add Cysteine 0 30 mg/g of protein

Dextrose %

Lipids 20% grams/kg/day over hours
Carnitine (10 mg/kg/day) added if lipids ordered

Sodium mEq/kg/day
Calculated 5000 mEq/liter

Potassium mEq/kg/day
Calculated 500 mEq/liter

Calcium 0 15 mEq/liter
 30 mEq/liter

Magnesium 0 5 mEq/liter

Phosphate 15 mmol/liter
(calculated from calcium dose)

Acetate/Chloride

4 Calculate (Updates Fields)

Minimal Chloride
 1:1 ratio
 Minimal Acetate

Amino Acid Calories: 8 kcal/kg/day Fat Calories: 20 kcal/kg/day
Dextrose Calories: 3.4 kcal/kg/day Total Calories: 31.4 kcal/kg/day
Lipid Rate: 1.6 ml/hr Lipid Volume: 10 ml/kg/day
Calculated minimum TPN Rate: 6.3 ml/hr
Calculated minimum TPN Volume: 152 ml/day
Calculated TPN Rate: 1.6 ml/hr
Calculated TPN Volume: 38 ml/day
Total Fluid Volume (TPN + Fat): 20 ml/kg/day

5 Submit Final Order OR Exit Without Ordering

Added Medications and Supplements

MVI-PEDIATRIC: 5 ml (wt >= 2.5 kg)

Neotrace 8 Selenium daily M TH

heparin 0 0.25 units/ml

Other Possible Additives

Vitamin K 0 1 mg/day

ramotidine (Pepcid) (mg/kg/day) 0 1 2

albumin (g/kg/day) 0 0.5 1

Special Instructions to Pharmacy: Copyright (C) 2005, Vanderbilt University Medical Center

FIGURE 10.21. NICU total parenteral nutrition (TPN) advisor provides complex interactive advice and performs various calculations.

Advisor system at LDS Hospital in Salt Lake City is described in Chapter 8. The LDS advisor analyzes patient data and laboratory results in order to determine likely pathogens, and then determines the optimal treatment for the patient, including factors such as patient allergies and local patterns of antimicrobial functions into its assessment. In WizOrder, the Web browser pop-up window is used to dynamically generate patient-specific advisory content.^{49,50} Figure 10.21 illustrates the WizOrder TPN ordering advisor for the neonatal intensive care unit (NICU).

Conclusion

It is critical that system developers, the technologists maintaining the system, and clinical experts collaborate in managing clinical systems during development. Implementing decision-support capabilities within clinical systems requires an understanding of the clinical significance of a proposed intervention, detailed knowledge of the intervention itself, and a good understanding of the workflows of the clinicians who will be affected by the intervention. The authors have described multiple mechanisms for delivering decision support within the context of CPOE systems using

Vanderbilt's WizOrder system for illustration. There are three important axes to consider: the role of decision support, when to intervene, and the method of intervention. Framing decision support in this manner may help both developers and clinical end users to understand how to tailor the system whenever new decision-support needs arise. This framework may also be useful when evaluating and reviewing decision support within CPOE systems.

Offering decision support within a CPOE system provides both clinical end users and institutional administrators with the opportunity to substantially change the way that an institution carries out its work, and to improve patient care processes in terms of quality and safety.

References

1. Kassirer JP, Gorry GA. Clinical problem solving: a behavioral analysis. *Ann Intern Med* 1978;89:245–255.
2. Covell DG, Uman GC, Manning PR. Information needs in office practice: are they being met? *Ann Intern Med* 1985;103:596–599.
3. Kassirer JP, Kopelman RI. Knowledge and clinical expertise. *Hosp Pract (Off Ed)* 1988;23:46, 53, 57 *passim*.
4. Jelovsek FR, Rittwage J, Pearse WH, Visscher HC. Information management needs of the obstetrician-gynecologist—a survey. *Obstet Gynecol* 1989;73(3 Pt 1):395–399.
5. Williamson JW, German PS, Weiss R, Skinner EA, Bowes F 3rd. Health science information management and continuing education of physicians. A survey of U.S. primary care practitioners and their opinion leaders. *Ann Intern Med* 1989;110:151–160.
6. Timpka T, Arborelius E. The GP's dilemmas: a study of knowledge need and use during health care consultations. *Methods Inf Med* 1990;29:23–29.
7. Berner ES, Shugerman AA. Needs assessment for diagnostic decision support systems (DDSS). *Proc Annu Symp Comput Appl Med Care* 1991:605–608.
8. Osheroff JA, Forsythe DE, Buchanan BG, Bankowitz RA, Blumenfeld BH, Miller RA. Physicians' information needs: analysis of questions posed during clinical teaching. *Ann Intern Med* 1991;114:576–581.
9. Forsythe DE, Buchanan BG, Osheroff JA, Miller RA. Expanding the concept of medical information: an observational study of physicians' information needs. *Comput Biomed Res* 1992;25:181–200.
10. Bowden VM, Kromer ME, Tobia RC. Assessment of physicians' information needs in five Texas counties. *Bull Med Libr Assoc* 1994;82:189–196.
11. Gorman PN, Ash J, Wykoff L. Can primary care physicians' questions be answered using the medical journal literature? *Bull Med Libr Assoc* 1994;82:140–146.
12. Lundeen GW, Tenopir C, Wermager P. Information needs of rural health care practitioners in Hawaii. *Bull Med Libr Assoc* 1994;82:197–205.
13. Wildemuth BM, de Blik R, Friedman CP, Miya TS. Information-seeking behaviors of medical students: a classification of questions asked of librarians and physicians. *Bull Med Libr Assoc* 1994;82:295–304.

14. Verhoeven AA, Boerma EJ, Meyboom-de Jong B. Use of information sources by family physicians: a literature survey. *Bull Med Libr Assoc* 1995;83:85–90.
15. McDonald CJ. Protocol-based computer reminders, the quality of care and the non-perfectability of man. *N Engl J Med* 1976;295:1351–1355.
16. McDonald CJ, Wilson GA, McCabe GP Jr. Physician response to computer reminders. *JAMA* 1980;244:1579–1581.
17. Tierney WM, Hui SL, McDonald CJ. Delayed feedback of physician performance versus immediate reminders to perform preventive care. Effects on physician compliance. *Med Care* 1986;24:659–666.
18. Tierney WM, McDonald CJ, Martin DK, Rogers MP. Computerized display of past test results. Effect on outpatient testing. *Ann Intern Med* 1987;107:569–574.
19. McDonald CJ, Tierney WM. Computer-stored medical records. Their future role in medical practice. *JAMA* 1988;259:3433–3440.
20. Tierney WM, McDonald CJ, Hui SL, Martin DK. Computer predictions of abnormal test results. Effects on outpatient testing. *JAMA* 1988;259:1194–1198.
21. McDonald CJ. Computers. *JAMA* 1989;261:2834–2836.
22. Tierney WM, Miller ME, McDonald CJ. The effect on test ordering of informing physicians of the charges for outpatient diagnostic tests. *N Engl J Med* 1990;322:1499–1504.
23. McDonald CJ, Tierney WM, Overhage JM, Martin DK, Wilson GA. The Regenstrief Medical Record System: 20 years of experience in hospitals, clinics, and neighborhood health centers. *MD Comput* 1992;9:206–217.
24. Tierney WM, Miller ME, Overhage JM, McDonald CJ. Physician inpatient order writing on microcomputer workstations. Effects on resource utilization. *JAMA* 1993;269:379–383.
25. McDonald CJ, Overhage JM. Guidelines you can follow and can trust. An ideal and an example [editorial; comment]. *JAMA* 1994;271:872–873.
26. Warner HR, Olmsted CM, Rutherford BD. HELP—a program for medical decision-making. *Comput Biomed Res* 1972;5:65–74.
27. Evans RS, Larsen RA, Burke JP, et al. Computer surveillance of hospital-acquired infections and antibiotic use. *JAMA* 1986;256:1007–1111.
28. Larsen RA, Evans RS, Burke JP, Pestotnik SL, Gardner RM, Classen DC. Improved perioperative antibiotic use and reduced surgical wound infections through use of computer decision analysis. *Infect Control Hosp Epidemiol* 1989;10:316–320.
29. Gardner RM, Golubjatnikov OK, Laub RM, Jacobson JT, Evans RS. Computer-critiqued blood ordering using the HELP system. *Comput Biomed Res* 1990; 23:514–528.
30. Pestotnik SL, Evans RS, Burke JP, Gardner RM, Classen DC. Therapeutic antibiotic monitoring: surveillance using a computerized expert system. *Am J Med* 1990;88:43–48.
31. Evans RS, Pestotnik SL, Classen DC, et al. Development of a computerized adverse drug event monitor. *Proc Annu Symp Comput Appl Med Care* 1991: 23–27.
32. Lepage EF, Gardner RM, Laub RM, Jacobson JT. Assessing the effectiveness of a computerized blood order “consultation” system. *Proc Annu Symp Comput Appl Med Care* 1991:33–37.

33. Classen DC, Evans RS, Pestotnik SL, Horn SD, Menlove RL, Burke JP. The timing of prophylactic administration of antibiotics and the risk of surgical-wound infection [see comments]. *N Engl J Med* 1992;326:281–286.
34. Evans RS, Pestotnik SL, Classen DC, Bass SB, Burke JP. Prevention of adverse drug events through computerized surveillance. *Proc Annu Symp Comput Appl Med Care* 1992:437–441.
35. Hales JW, Gardner RM, Huff SM. Integration of a stand-alone expert system with a hospital information system. *Proc Annu Symp Comput Appl Med Care* 1992:427–431.
36. Evans RS, Classen DC, Stevens LE, et al. Using a hospital information system to assess the effects of adverse drug events. *Proc Annu Symp Comput Appl Med Care* 1993:161–165.
37. Gardner RM, Christiansen PD, Tate KE, Laub MB, Holmes SR. Computerized continuous quality improvement methods used to optimize blood transfusions. *Proc Annu Symp Comput Appl Med Care* 1993:166–170.
38. Evans RS, Classen DC, Pestotnik SL, Lundsgaarde HP, Burke JP. Improving empiric antibiotic selection using computer decision support. *Arch Intern Med* 1994;154:878–884.
39. White KS, Lindsay A, Pryor TA, Brown WF, Walsh K. Application of a computerized medical decision-making process to the problem of digoxin intoxication. *J Am Coll Cardiol* 1984;4:571–576.
40. Halpern NA, Thompson RE, Greenstein RJ. A computerized intensive care unit order-writing protocol. *Ann Pharmacother* 1992;26:251–254.
41. Dambro MR, Weiss BD, McClure CL, Vuturo AF. An unsuccessful experience with computerized medical records in an academic medical center. *J Med Educ* 1988;63:617–623.
42. Williams LS. Microchips versus stethoscopes: Calgary hospital, MDs face off over controversial computer system. *CMAJ* 1992;147:1534–1540, 1543–1544, 1547.
43. Massaro TA. Introducing physician order entry at a major academic medical center: II. Impact on medical education. *Acad Med* 1993;68:25–30.
44. Massaro TA. Introducing physician order entry at a major academic medical center: I. Impact on organizational culture and behavior. *Acad Med* 1993; 68:20–25.
45. Geissbuhler A, Miller RA. A new approach to the implementation of direct care-provider order entry. *Proc AMIA Annu Fall Symp* 1996:689–693.
46. Stead WW, Borden R, Bourne J, et al. The Vanderbilt University fast track to IAIMS: transition from planning to implementation. *J Am Med Inform Assoc* 1996;3:308–317.
47. Geissbuhler A, Grande JF, Bates RA, Miller RA, Stead WW. Design of a general clinical notification system based on the publish-subscribe paradigm. *Proc AMIA Annu Fall Symp* 1997:126–130.
48. Geissbuhler A, Miller RA, Stead WW. The clinical spectrum of decision-support in oncology with a case report of a real world system. In: Elpida T. Keravnou, Catherine Garbay, et al. eds. *Artificial Intelligence Medicine, 6th Conference on Artificial Intelligence in Medicine in Europe, AIME'97, Grenoble, France, March 23–26, 1997, Proceedings. Lecture Notes in Computer Science 1211 Springer* 1997.
49. Heusinkveld J, Geissbuhler A, Sheshelidze D, Miller R. A programmable rules engine to provide clinical decision support using HTML forms. *Proc AMIA Symp* 1999:800–803.

50. Starmer JM, Talbert DA, Miller RA. Experience using a programmable rules engine to implement a complex medical protocol during order entry. *Proc AMIA Symp* 2000;829–832.
51. Stead WW, Bates RA, Byrd J, Giuse DA, Miller RA, Shultz EK. Case study: the Vanderbilt University Medical Center information management architecture. In: Van De Velde R, Degoulet P, eds. *Clinical information systems: a component-based approach*. New York: Springer-Verlag; 2003.
52. Neilson EG, Johnson KB, Rosenbloom ST, et al. The impact of peer management on test-ordering behavior. *Ann Intern Med* 2004;141:196–204.
53. Potts AL, Barr FE, Gregory DF, Wright L, Patel NR. Computerized physician order entry and medication errors in a pediatric critical care unit. *Pediatrics* 2004;113(1 Pt 1):59–63.
54. Rosenbloom ST, Chiu KW, Byrne DW, Talbert DA, Neilson EG, Miller RA. Interventions to regulate ordering of serum magnesium levels: an unintended consequence of decision support. *J Am Med Inform Assoc* 2005;12:546–553.
55. Rosenbloom ST, Geissbuhler AJ, Dupont WD, et al. Effect of CPOE user interface design on user-initiated access to educational and patient information during clinical care. *J Am Med Inform Assoc* 2005;12:458–473.
56. Teich JM, Spurr CD, Schmiz JL, O'Connell EM, Thomas D. Enhancement of clinician workflow with computer order entry. *Proc Annu Symp Comput Appl Med Care* 1995:459–463.
57. McDonald CJ, Overhage JM, Tierney WM, et al. The Regenstrief Medical Record System: a quarter century experience. *Int J Med Inform* 1999;54:225–253.
58. van Wijk MA, van der Lei J, Mosseveld M, Bohnen AM, van Bommel JH. Assessment of decision support for blood test ordering in primary care: a randomized trial. *Ann Intern Med* 2001;134:274–281.
59. Kalmeijer MD, Holtzer W, van Dongen R, Guchelaar HJ. Implementation of a computerized physician medication order entry system at the Academic Medical Centre in Amsterdam. *Pharm World Sci* 2003;25:88–93.
60. Payne TH, Hoey PJ, Nichol P, Lovis C. Preparation and use of preconstructed orders, order sets, and order menus in a computerized provider order entry system. *J Am Med Inform Assoc* 2003;10:322–329.
61. Oppenheim MI, Vidal C, Velasco FT, et al. Impact of a computerized alert during physician order entry on medication dosing in patients with renal impairment. *Proc AMIA Symp* 2002:577–581.
62. Kuperman GJ, Teich JM, Tanasijevic MJ, et al. Improving response to critical laboratory results with automation: results of a randomized controlled trial. *J Am Med Inform Assoc* 1999;6:512–522.
63. Stead WW, Miller RA, Musen MA, Hersh WR. Integration and beyond: linking information from disparate sources and into workflow. *J Am Med Inform Assoc* 2000;7:135–145.
64. Chertow GM, Lee J, Kuperman GJ, et al. Guided medication dosing for inpatients with renal insufficiency. *JAMA* 2001;286:2839–2844.
65. Murff HJ, Kannry J. Physician satisfaction with two order entry systems. *J Am Med Inform Assoc* 2001;8:499–509.
66. Kuperman GJ, Teich JM, Gandhi TK, Bates DW. Patient safety and computerized medication ordering at Brigham and Women's Hospital. *Jt Comm J Qual Improv* 2001;27:509–521.

67. Bizovi KE, Beckley BE, McDade MC, et al. The effect of computer-assisted prescription writing on emergency department prescription errors. *Acad Emerg Med* 2002;9:1168–1175.
68. Barlow IW, Flynn NA, Britton JM. The Basingstoke Orthopaedic Database: a high quality accurate information system for audit. *Ann R Coll Surg Engl* 1994;76(6 Suppl):285–287.
69. Apkon M, Singhaviranon P. Impact of an electronic information system on physician workflow and data collection in the intensive care unit. *Intensive Care Med* 2001;27:122–130.
70. Teich JM, Glaser JP, Beckley RF, et al. The Brigham integrated computing system (BICS): advanced clinical systems in an academic hospital environment. *Int J Med Inform* 1999;54:197–208.
71. Bates DW, Kuperman G, Teich JM. Computerized physician order entry and quality of care. *Qual Manag Health Care* 1994;2:18–27.
72. Bates DW. Using information technology to reduce rates of medication errors in hospitals. *BMJ* 2000;320:788–791.
73. Teich JM, Merchia PR, Schmitz JL, Kuperman GJ, Spurr CD, Bates DW. Effects of computerized physician order entry on prescribing practices. *Arch Intern Med* 2000;160:2741–2747.
74. Bates DW, Cohen M, Leape LL, Overhage JM, Shabot MM, Sheridan T. Reducing the frequency of errors in medicine using information technology. *J Am Med Inform Assoc* 2001;8:299–308.
75. Tierney WM, Overhage JM, McDonald CJ. Computerizing guidelines: factors for success. *Proc AMIA Annu Fall Symp* 1996:459–462.
76. Chin HL, Wallace P. Embedding guidelines into direct physician order entry: simple methods, powerful results. *Proc AMIA Symp* 1999:221–225.
77. Evans RS, Pestotnik SL, Classen DC, et al. A computer-assisted management program for antibiotics and other anti-infective agents. *N Engl J Med* 1998;338:232–238.
78. Maviglia SM, Zielstorff RD, Paterno M, Teich JM, Bates DW, Kuperman GJ. Automating complex guidelines for chronic disease: lessons learned. *J Am Med Inform Assoc* 2003;10:154–165.
79. Bates DW, Kuperman GJ, Rittenberg E, et al. A randomized trial of a computer-based intervention to reduce utilization of redundant laboratory tests. *Am J Med* 1999;106:144–150.
80. Bates DW, Kuperman GJ, Jha A, et al. Does the computerized display of charges affect inpatient ancillary test utilization? *Arch Intern Med* 1997;157:2501–2508.
81. Dexter PR, Wolinsky FD, Gramelspacher GP, et al. Effectiveness of computer-generated reminders for increasing discussions about advance directives and completion of advance directive forms. A randomized, controlled trial. *Ann Intern Med* 1998;128:102–110.
82. Gamble ER, McDonald PJ, Lichstein PR. Knowledge, attitudes, and behavior of elderly persons regarding living wills. *Arch Intern Med* 1991;151:277–280.
83. Shojania KG, Yokoe D, Platt R, Fiskio J, Ma'luf N, Bates DW. Reducing vancomycin use utilizing a computer guideline: results of a randomized controlled trial. *J Am Med Inform Assoc* 1998;5:554–562.
84. Fischer MA, Solomon DH, Teich JM, Avorn J. Conversion from intravenous to oral medications: assessment of a computerized intervention for hospitalized patients. *Arch Intern Med* 2003;163:2585–2589.

85. Harpole LH, Khorasani R, Fiskio J, Kuperman GJ, Bates DW. Automated evidence-based critiquing of orders for abdominal radiographs: impact on utilization and appropriateness. *J Am Med Inform Assoc* 1997;4:511–521.
86. Bates DW, Kuperman GJ, Wang S, et al. Ten commandments for effective clinical decision support: making the practice of evidence-based medicine a reality. *J Am Med Inform Assoc* 2003;10:523–530.
87. Overhage JM, Tierney WM, Zhou XH, McDonald CJ. A randomized trial of “corollary orders” to prevent errors of omission. *J Am Med Inform Assoc* 1997;4:364–375.
88. Horsky J, Kaufman DR, Patel VL. Computer-based drug ordering: evaluation of interaction with a decision-support system. *Medinfo* 2004;11(Pt 2):1063–1067.
89. Kucher N, Koo S, Quiroz R et al. Electronic alerts to prevent venous thromboembolism among hospitalized patient. *N Engl J Med* 2005;352:969–977.
90. Payne TH. The transition to automated practitioner order entry in a teaching hospital: the VA Puget Sound experience. *Proc AMIA Symp* 1999:589–593.
91. Lovis C, Chapko MK, Martin DP, et al. Evaluation of a command-line parser-based order entry pathway for the Department of Veterans Affairs electronic patient record. *J Am Med Inform Assoc* 2001;8:486–498.
92. Ash JS, Berg M, Coiera E. Some unintended consequences of information technology in health care: the nature of patient care information system-related errors. *J Am Med Inform Assoc* 2004;11:104–112.
93. Gainer A, Pancheri K, Zhang J. Improving the human computer interface design for a physician order entry system. *AMIA Annu Symp Proc* 2003:847.

11

Decision Support for Patients

HOLLY B. JIMISON, PAUL P. SHER, and JENNIFER J.B. JIMISON

This chapter is designed to introduce the concept of computer-based decision support systems for patients. With the rapid growth of computing technology available to consumers and the virtual explosion of health information available on the World Wide Web, patient decision aids and computer-based health interventions are now a more common part of routine medical care. The new field of consumer health informatics deals with “developing and evaluating methods and applications to integrate consumer needs and preferences into information management systems in clinical practice, education, and research.”¹ This technology, both hardware and software, is part of a growing trend toward empowering consumers to take a more active role in their own health care and to provide the necessary information to enhance their decision making. Today, more than ever, consumers are using information technology as an important supplement to the information provided by healthcare professionals in the course of clinical encounters.

Role of Consumer Health Informatics in Patient Care

Research studies have shown that access to health information can enable patients to be more active participants in the treatment process, leading to better medical outcomes.²⁻⁵ Health education is an important aspect of doctor-patient communication. Patients report that they want to be informed about their medical condition,^{6,7} and the process of sharing information enhances the doctor-patient relationship. The rapid growth of consumer health software and materials on the Web has facilitated patient participation in their health care and decision making. These systems have been developed to assist patients with informed consent,⁸ as well as coping and decision-making skills.^{9,10} Involvement in one’s medical care also involves the concepts of patient empowerment and self-efficacy.

Empowerment and Self-Efficacy

Empowerment and self-efficacy are closely linked concepts. In general, empowerment can be thought of as the process that enables people to “own” their own lives and have control over their destiny. It is closely related to health outcomes in that powerlessness has been shown to be a broad-based risk factor for disease. Studies demonstrate that patients who feel “in control” in a medical situation have better outcomes than those who feel “powerless.”^{11–13}

Similarly, self-efficacy is a patient’s level of confidence that he or she can perform a specific task or health behavior in the future. Several clinical studies have shown self-efficacy to be the variable most predictive of improvements in patients’ functional status.^{14–21} For example, in a study of functional status after bypass surgery, self-efficacy explained more variability in functional status outcomes than did measures of disease severity, functional capacity, comorbidity, or preoperative functioning.²² Additionally, in a study on patients with rheumatoid arthritis, the degree of perceived self-efficacy was correlated with reduced pain and joint inflammation and improved psychosocial functioning.¹⁶ In cancer patients, a strong positive correlation was found between self-efficacy and quality of life and mood.²³ In the prevention area, perceived self-efficacy was shown to play a significant role in smoking cessation relapse rate, control of eating and weight, and adherence to general preventive health programs.²⁴

Given the strong influence of empowerment and self-efficacy on health outcomes, it is important to incorporate a focus on these concepts when designing systems for patient use. The feeling of empowerment and self-efficacy can be enhanced, for instance, by online support groups where patients are able to connect, communicate, and engage in problem solving with others who have similar medical problems. This has been demonstrated by the Comprehensive Health Enhancement and Support System (CHESS) in women with breast cancer and patients with AIDS.^{9,10,25} An important measure of the success of health information systems is how well they promote empowerment and self-efficacy for patients.

Incorporating Patient Preferences

As medical care increasingly focuses on chronic disease, it is especially important that patient preferences regarding the long-term effects of their medical care be taken into account. For patients to be adequately informed to make decisions regarding their medical care, it is important that they obtain information about the quality of life associated with the possible medical outcomes of these decisions. Yet the reliable assessment of a patient’s preferences and risk attitudes for clinical outcomes is probably the weakest link in most clinical decision making. Recent efforts to explore the use of computers in communication about health outcomes, and

in assessing patients' preferences for various health outcomes, have started to address these issues.²⁶⁻²⁸ Information on patient preferences is important for tailoring information to patients and for providing decision support.²⁶ Tailored information has been found to be more effective in providing consumer information²⁹ and is preferred by patients.³⁰ In addition to differences in preferences for health outcomes, patients differ in the degree to which they choose to be involved in decision making. Research confirms that age (younger), gender (females), and education level (more) are strong predictors of the desire to be involved in medical decisions. There is also a higher desire to be involved in medical decisions that appear to require less medical expertise, such as a knee injury, as opposed to a cancerous growth.³⁰

The Computer as a Health Information Medium

There has been an increase in research devoted to testing the effectiveness of various formats and types of media for conveying health information to consumers.³¹⁻³⁵ These studies tend to show that video and slides are educationally more effective than books and audiotapes. Computer approaches have the additional advantages of interactivity, providing feedback in the learning process and the ability to tailor information to the individual patient. However, in many cases, more research is required to demonstrate the effectiveness of computer approaches. In addition, designers of systems for patients have not always been sufficiently sensitive to human-computer interface issues. The design of a system for general health education for patients requires specifications that meet a variety of needs. Table 11.1 outlines the design guidelines for a consumer health information system.³⁶

Health Information and Decision Support Systems for Patients

The number of commercial computer and Web-based products to support patients' health information needs is expanding rapidly. The information and decision aids range from general home healthcare reference information to symptom management and diagnostic decision support. There has been a dramatic upsurge in consumers' use of the Web to acquire health information. Physicians, clinics, hospitals, and insurers are all redefining their business practices to incorporate the Internet and Web delivery systems. In addition, in 2004 the Pew Survey on the Internet and American Life found that 55% of Internet users have used the Web to seek health information.³⁷ Whereas many computer tools for patients, in the past, were

TABLE 11.1. Design guidelines for a consumer health information system.

Intuitive Interface

- Graphical metaphors easily understood by the general populace
- Designed for use by naïve, untrained users
- Online help available at every stage
- Immediate word definitions available in every application

Complete Coverage/Coordination

- Single location for information on disease and health concerns
- Coordinated with routine medical care

Hierarchical Presentation

- Simple summary information presented first
- More detail and complexity available as desired
- Guided movement through databases
- User requests anticipated, presearch to improved speed

Presentation Tailored to the Individual

- Material presented appropriate for the assessed reading level
- Material presented appropriate for education and medical expertise
- Material presented in a culturally sensitive manner
- Material presented in the appropriate language
- Material tailored to history and assessed patient-specific health risks
- Patient preferences incorporated

Facilitate Quality Decision Making

- Health outcomes information included
- Patient preferences on health outcomes incorporated
- Summary of tailored decision support information

Option for Printout

- Ability for the patient to have material to take home/share with family included

delivered via CD-ROM or videodisc, current technology with broadband access, makes Web delivery far less expensive and more easily accessible by most patients. The following sections describe the various types of health information and decision support applications available for patients and their families.

General Health References

There are several publishers of Web-based general health encyclopedic reference materials. These materials are typically alphabetically arranged and searchable. Topics include prevention, diseases, treatments, and procedures. Oftentimes, content creators and publishers, such as A.D.A.M.³⁸ and Healthwise³⁹ will license their content to several other clients that, in turn, deliver Web material for patients. These clients may be health insurance companies, health portals, such as WebMD⁴⁰, or general Web portals with health services, such as Yahoo! and AOL. The U.S. Federal Government also supports a Web service for consumers on their Healthfinder site.⁴¹

The material on this site is a compendium of information available from other government sites, but written at a lay level. The government also provides MedlinePlus to assist patients in searching for reliable health information. Several academic medical centers, such as Mayo Clinic, have also created their own consumer-based Web sites based on locally created information.⁴²

Web sites that present a full complement of health information will also typically organize information by age, e.g., child's health, women's health, men's health, or senior health. Alternate organizational schemes that users find helpful include separating information by disease, and then linking possible symptoms, diagnostic procedures, and treatments within that cluster. Sites such as WebMD⁴⁰ have also found it useful to have a separate section with information for newly diagnosed patients. More detailed information on specific diseases or conditions, is often available from societies or groups specializing in a topic. Finally, online medical dictionaries, disease-specific discussion boards, and "ask-an-expert" services are also often found as components of health portal sites.

Drug Information

Information about prescriptions is most commonly obtained from content providers and publishers that specialize in just that feature. Examples of searchable drug databases for patient use are RxList,⁴³ DrugInfoNet,⁴⁴ and RxMed.⁴⁵ These databases include general and interaction information on a large number of prescription and over-the-counter drugs. Additionally, some of the health portals offer a Web tool that specifically checks for drug-drug interactions for a particular patient's set of prescriptions. Multum⁴⁶ is one such company that provides this database and algorithms for these Web sites.

Diagnostic Decision Support

Some of the health portals that offer general reference and drug information also offer interactive tools to assist patients in health assessment, symptom management, and limited diagnostic information (usually in preparation for shared decision making in an office visit). Health risk appraisals usually take the form of a questionnaire with questions on family history and health behaviors. After completion, patients receive a tailored printout with a summary of results that may help them prioritize their health goals. Companies, such as WellMed,⁴⁷ will provide this service for organizations with health portals. This information may then be linked to a personal health record and shared with one's clinician.

Another example of online self-assessment using a questionnaire style format is Mayo Clinic's depression test.⁴² It is embedded in a page with links to further information on depression. These types of assessments allow the

patient to know when to pursue diagnostic advice from a health care professional and when to seek treatment.

Many of the health portals also offer calculator style tools to help patients manage their health. For example, after entering height and weight, patients can obtain their body mass index. Pregnancy calculations and target heart rate calculations are also amenable to this approach.

Occasionally, the health Web sites will offer diagnostic aids for patients. However, there has been some reluctance to offer advice that is overly specific. The usual approach on the health sites that offer symptom-based diagnosis is to assess a symptom or two and then present a list of possible causes, with links to further reading. The Mayo Clinic Web site for patients offers what they call Health Decision Guides⁴² for a limited number of diseases and conditions. The guides consist of background material on the condition, how the condition is diagnosed, treatment options with thorough descriptions, and the pros and cons of each option. Within their site, the background material is supplemented with video clips on the Web. This goal of this approach is not to provide the patient with a diagnosis or specific recommendation, but to prepare the patient to be an informed participant in making the treatment decision during the next visit to the clinician.

The Foundation for Informed Medical Decision Making (FIMDM) has taken the decision assistance even further. They focus on treatment decisions where patient preferences on health outcomes are important.⁴⁸ They use video (in some cases interactive) to convey to patients what it might be like to live with possible future outcomes. With the background on prognosis and an understanding of the related health outcomes, patients will be better prepared to participate in shared decision making with their clinician when treatment goals are set. FIMDM provides tools for breast cancer, prostate cancer, knee osteoarthritis, back pain, coronary artery disease, and reproductive conditions. Treatment decisions for chronic conditions such as these are often quite sensitive to patient preferences, and it is important to have tools that facilitate patient participation in decision making.

It is interesting to note that health books and computer applications, for patients, on CD-ROMs, are likely to have more explicit diagnostic information with recommendations than what is currently seen on the Web. For example, the American Medical Association's Family Medical GuideTM is a program consisting of seven modules: (1) diseases, disorders, and conditions; (2) an atlas of the body; (3) symptoms and self-diagnosis; (4) your healthy body; (5) injuries and emergencies; (6) diagnostic imaging techniques; and (7) caring for the sick. The program's diagnostic decision support section consists of a large number of symptom flow charts that are organized alphabetically, or can be accessed by either pain-site diagrams or body system diagnosis. The diagnostic symptom charts are flow diagrams in which each question is read to the user by the computer. A "yes" or "no" answer to the question directs the user through the flow diagram to the next question, leading to a patient-specific recommendation.

The Home Medical Advisor Pro™ CD-ROM software has a symptom analysis program for single symptoms, and a symptom complex analysis program for multiple symptoms.⁴⁹ Users are asked a series of questions to further characterize their symptoms. The analysis of the multiple symptom case uses rule-based algorithms to identify diseases that match the symptoms and then restructures the list based on disease likelihood. The analysis lists possible diseases categorized in two ways. First, the number of symptoms that match with a particular disease is listed, starting with diseases with the most matches. Second, the probabilities appear in parentheses following the diagnosis (expressed as very common, common, rare, and very rare). The Home Medical Advisor Pro™ also has a drug interaction tool that can provide users with feedback on 500,000 possible interactions.

The two patient software packages, Medical HouseCall™ and Pediatric HouseCall™, have sophisticated diagnostic features. The algorithms were derived from a diagnostic and treatment expert system for clinicians, known as Iliad™, developed in Utah by a team from Applied Medical Informatics, and from the Department of Medical Informatics and the University of Utah.⁴⁹ The knowledge base and inference engine were restructured for the patient software systems to accommodate consumer queries and provide lay answers. After a user inputs answers to a series of questions from the software, the system presents a ranked order of disease likelihoods based on symptoms and Bayesian analysis. The software points out the remaining uncertainty of this list, since selecting an actual diagnosis would require physical examination and possible tests. In adapting the Iliad knowledge base for home use with patients, the developers eliminated the physical exam findings and lab tests. The vocabulary was translated into “consumer language.” Additionally, the diagnostic probabilities were adjusted not to exceed 70%, the estimated contribution of historical information to making a diagnosis. As with the other diagnosis programs for patients, Medical HouseCall™ and Pediatric HouseCall™ also have drug interaction modules.

Helping Patients Judge the Quality of Health Information

Judging the quality of health materials on the Web or in software packages is particularly challenging for consumers. For books or CD-ROMs, consumers may have an indication of the credibility of the publisher. However, there are minimal monetary and skill barriers to creating Web sites, and it is fairly easy to make a site look quite professional and indistinguishable from those of larger, well established organizations. Not all sites are “peer reviewed,” published, or created by professionals with expertise in the topic covered. Because the quality of health information is so critical for consumers, several organizations have created guidelines for judging the

quality of information on the Web for consumers.⁵⁰⁻⁵² Some of the criteria included in all of these guidelines are topical relevance, currency of the information, accuracy, and authoritativeness or objectivity.

From the consumer's point of view, topical relevance is certainly important when assessing the usefulness and quality of a Web site or computer application. The relevance of a site is context-specific and depends on the particular question an individual consumer has in mind. To find appropriate materials, sites must be clearly organized and/or have intelligent search functions. In addition, the relevance of the material depends on the degree to which it is tailored to the individual and is appropriate to their specific needs. Most health material on the Web is generic and not interactively tailored to individuals. This basically replicates what could be found in a textbook or brochure. The final aspect of relevance to an individual has to do with whether the material is action oriented, and either helps the consumer make a healthcare decision that may lead to an action or promotes health behavior change.

Currency or the timeliness of information is an important consideration. It is often difficult to have a generalized policy on how often health materials need to be updated. However, most professional sites ensure at least quarterly review of all materials. Consumers may judge the currency of Web site information by looking for date stamps or a notice of date of creation and/or update. It is important to note that some Web sites use algorithms to automatically update their time stamp even if the material has not been changed or even reviewed, giving the impression of the information being current. Responding to the difficulty that consumers are likely to have in judging these aspects of Web site quality, the Health on the Net (HON) Foundation⁵² has promoted an ethical code of conduct and a set of standards for Web site developers to ensure the reliability of medical and health information available on the World Wide Web. Consumer health sites that display an HON certificate signify that they are in compliance with the HON code of conduct and standards. Providing health information and interventions over the Internet is becoming an increasingly important component of health care. Ensuring that the materials are unbiased, accurate, relevant, and timely is fundamental to providing quality health care.

Patient Access to Decision Support Systems

As the demand for more health information and decision support grows, the need for wider availability of these systems becomes even more important. Today, these systems can be found in a variety of settings and in a variety of forms. In addition to consumers searching the Web at home, public access computer systems can be found in public libraries, health resource centers, worksites, schools, and community centers. Different

systems may require quite different physical locations. For instance, many patients are uncomfortable exploring sensitive health information in a public space.

There are many factors that influence the health information seeking behavior of patients. As documented by Harris, these factors include demographic divisions such as age, gender, disability, race and ethnicity, and socioeconomic status.⁵³ Research indicates that these demographic variables can predict differences in the amount and type of health information that patients want. Whereas some patients may not seek much information, for many of those who do desire information, serious barriers to the use of these systems still exist.

A lack of reading ability is a functional barrier affecting use of computer systems. Approximately one out of five Americans is functionally illiterate, reading at or below the fifth-grade level. Most studies on the comprehension of health education handouts, typically show that only half of the patients are able to comprehend written health materials.⁵⁴⁻⁵⁶ Studies confirmed that patients' reading levels were well below what was needed to understand standard health brochures.⁵⁷ In developing health information for patients, one cannot assume that a patient who has completed a certain grade level in school can read at the corresponding level. Numerous studies on literacy and readability confirm the widespread problem of low literacy skills.⁵⁸⁻⁶⁰ Health materials should be written at least three grade levels lower than the average educational level of the target population.⁶¹ Text characteristics also play an important role in comprehension and retention of material. Organization and clarity need to be considered in creating educational materials.⁶² Computers with multimedia capabilities can correct some of these problems by conveying information through video, audio and graphics that would normally be presented as written text. These systems can also be adapted for multiple foreign languages.

In addition to language and literacy issues, an area that is often overlooked relates to the cultural issues associated with health information-seeking behavior and the willingness to use computers to access health information. Most developers have not invested the time to develop systems that are culturally and linguistically relevant to diverse populations.

Finally, the question of who will pay for the access and use of technologies for consumer health information is still an unresolved issue. Educational and socioeconomic factors still determine access to computers and information technologies. Younger, more affluent, and well educated patients are more likely to have access to home computers, diagnostic software, and Internet access. The poor and socioeconomically disadvantaged already have worse health outcomes and worse access to medical care. Special effort is required to ensure ease of access and ease of use of health information systems so as to not further disadvantage the very people who have the greatest need for these resources.

The Future of Decision Support Systems for Patients

Advances in communications and information processing technology are certainly changing the way in which medicine is practiced, with dramatic impact on how patients are beginning to receive their health information and interact with the medical care system. There has also been a shift toward consumers becoming empowered participants and assuming a more active role in their medical care decisions, through increased and more effective access to healthcare information and decision tools. The developers of computer applications for patients have pushed the field of consumer health informatics forward with many innovative systems. However, to achieve significant improvements in the quality of care and health outcomes, researchers and system developers need to focus on bringing the knowledge gained from previous work in health education and behavior change into the design of new systems. This is a rapidly developing field, with significant innovations in the commercial sector. However, research in several areas is needed to move the field forward in providing real benefits to patients' health outcomes and in showing the effectiveness of the systems to purchasers of health care. The criteria for evaluating computer-based decision support systems for patients are similar to the criteria for physician systems, namely accuracy and effectiveness.⁶³ However, the rapid deployment of these systems, in an ever changing medical care environment, makes critical evaluation of consumer health information systems extremely difficult. Web sites change daily, and access to one system usually means increased access to many others. It is important to understand the potential effectiveness of investments in this area. Careful needs assessment before system development, usability testing during development, controlled clinical trials, and studies of use and outcomes in natural settings are all critical to our understanding of how to best provide health information and decision assistance to patients.

References

1. Eysenbach G. Recent advances: consumer health informatics. *BMJ* 2000;320:1713-1716.
2. Brody DS, Miller SM, Lerman CE, Smith DG, Caputo GC. Patient perception of involvement in medical care: relationship to illness attitudes and outcomes. *J Gen Intern Med* 1989;4:506-511.
3. Greenfield S, Kaplan S, Ware JE Jr. Expanding patient involvement in care: effects on patient outcomes. *Ann Intern Med* 1985;102:520-528.
4. Korsch BM. What do patients and parents want to know? What do they need to know? *Pediatrics* 1984;74(5 Pt 2):917-919.
5. Mahler HI, Kulik JA. Preferences for health care involvement, perceived control and surgical recovery: a prospective study. *Soc Sci Med* 1990;31:743-751.
6. Ende J, Kazis L, Ash A, Moskowitz MA. Measuring patients' desire for autonomy: decision making and information-seeking preferences among medical patients. *J Gen Intern Med* 1989;4:23-30.

7. Waitzkin H. Doctor-patient communication: clinical implications of social scientific research. *JAMA* 1984;252:2441-2446.
8. Llewellyn-Thomas HA, Thiel EC, Sem FW, Woermke DE. Presenting clinical trial information: a comparison of methods. *Patient Educ Couns* 1995;25:97-107.
9. Gustafson DH, Bosworth K, Hawkins RP, Boberg EW, Bricker E. CHESS: a computer-based support system for providing information, referrals, decision support and social support to people facing medical and other health-related crises. *Proc Annu Symp Comput Appl Med Care* 1992:161-165.
10. Pingree S, Hawkins RP, Gustafson DH, et al. Will HIV-positive people use an interactive computer system for information and support? A study of CHESS in two communities. *Proc Annu Symp Comput Appl Med Care* 1993: 22-26.
11. Peterson C, Stunkard AJ. Personal control and health promotion. *Soc Sci Med* 1989;28:819-828.
12. Cassileth B, Zupkis R, Sutton-Smith K, March V. Information and participation preferences among cancer patients. *Ann Intern Med* 1980;92:832-836.
13. Israel BA, Sherman SJ. Social support, control and the stress process. In: Glanz K, Lewis FM, Rimer BK, eds. *Health behavior and health education: theory, research and practice*. San Francisco: Jossey-Bass; 1990.
14. Mullen PD, Laville EA, Biddle AK, Lorig K. Efficacy of psychoeducational interventions on pain, depression, and disability in people with arthritis: a meta-analysis. *J Rheumatol* 1987;14(Suppl 15):33-39.
15. Maibach E, Flora J, Nass C. Changes in self-efficacy and health behavior in response to a minimal contact community health campaign. *Health Commun* 1991;3:1-15.
16. Lorig K, Chastain RL, Ung E, Shoor S, Holman HR. Development and evaluation of a scale to measure perceived self-efficacy in people with arthritis. *Arthritis Rheum* 1989;32:37-44.
17. Holman H, Lorig K. Patient education in the rheumatic diseases—pros and cons. *Bull Rheum Dis* 1987;37(5):1-8.
18. Bandura A. Self-efficacy: towards a unifying theory of behavioral change. *Psychol Rev* 1977;84:191-215.
19. O'Leary A, Shoor S, Lorig K, Holman HR. A cognitive-behavioral treatment for rheumatoid arthritis. *Health Psychol*, 1988;7:527-544.
20. Feste C, Anderson RM. Empowerment: from philosophy to practice. *Patient Educ Couns* 1995;26:139-144.
21. Anderson RM, Funnell MM, Butler PM, Arnold MS, Fitzgerald JT, Feste CC. Patient empowerment. Results of a randomized trial. *Diabetes Care* 1995;18: 943-949.
22. Allen JK, Becker DM, Swank RT. Factors related to functional status after coronary artery bypass surgery. *Heart Lung* 1990;19:337-343.
23. Cunningham AJ, Lockwood GA. A relationship between perceived self-efficacy and quality of life in cancer patients. *Patient Educ Couns* 1991;17:71-78.
24. O'Leary A. Self-efficacy and health. *Behav Res Ther* 1985;23:437-451.
25. Gustafson DH, Hawkins RP, Boberg EW, Bricker E, Pingree S, Chan CL. The use and impact of computer support system for people living with AIDS and HIV infection. *Proc Annu Symp Comput Appl Med Care* 1994:604-608.
26. Jimison HB, Henrion M. Hierarchical preference models for patients with chronic disease. *Med Decis Making* 1992;7:351.

27. Goldstein MK, Clarke AE, Michelson D, Garber AM, Bergen MR, Lenert LA. Developing and testing a multimedia presentation of a health-state description. *Med Decis Making* 1994;14:336–344.
28. Lenert LA, Sturley A, Watson ME. iMPACT3: Internet-based development and administration of utility elicitation protocols. *Med Decis Making* 2002;22:464–474.
29. Skinner CS, Strecher VJ, Hoppers H. Physicians' recommendations for mammography: do tailored messages make a difference? *Am J Public Health* 1994;84:43–49.
30. Thompson SC, Pitts IS, Schwankovsky L. Preferences for involvement in medical decision-making: situational and demographic influences. *Patient Educ Couns* 1993;22:133–140.
31. Funnell MM, Donnelly MB, Anderson RM, Johnson PD, Oh MS. Perceived effectiveness, cost, and availability of patient education methods and materials. *Diabetes Educ* 1992;18:139–145.
32. Alterman AI, Baughman TG. Videotape versus computer interactive education in alcoholic and nonalcoholic controls. *Alcohol Clin Exp Res* 1991;15:39–44.
33. Gillispie MA, Ellis LB. Computer-based patient education revisited. *J Med Syst* 1993;17:119–125.
34. Consoli SM, Ben Said M, Jean J, Menard J, Plouin PF, Chatellier G. Benefits of a computer-assisted education program for hypertensive patients compared with standard education tools. *Patient Educ Couns* 1995;26:343–347.
35. Skinner CS, Siegfried JC, Kegler MC, Strecher VJ. The potential of computers in patient education. *Patient Educ Couns* 1993;22:27–34.
36. Jimison HB, Sher PP. Presenting clinical and consumer data to patients. In Chapman GB, Sonnenberg FA, eds. *Decision making in health care: theory, psychology, and applications*. New York: Cambridge University Press; 2000.
37. Pew Internet & American Life, Online health care revolution: how the Web helps Americans take better care of themselves, www.pewinternet.org/PPF/r/26/report_display.asp; accessed November 2004.
38. A.D.A.M. www.adam.com; accessed November 2004.
39. Healthwise, www.healthwise.com; accessed November 2004.
40. WebMD, www.webmd.com; accessed November 2004.
41. Healthfinder, www.healthfinder.gov; accessed November 2004.
42. MayoClinic.com, www.mayoclinic.com; accessed November 2004.
43. RxList, www.rxlist.com; accessed November 2004.
44. DrugInfoNet, www.druginfonet.com; accessed November 2004.
45. RxMed, www.rxmed.com; accessed November 2004.
46. Multum, www.multum.com/Lexicon.htm; accessed November 2004.
47. Wellmed, www.wellmed.com; accessed November 2004.
48. Foundation for Informed Medical Decision Making, www.FIMDM.org; accessed November 2004.
49. Bouhaddou O, Warner H. An interactive patient information and education system (Medical HouseCall) based on a physician expert system (Iliad). *MedInfo* 1995;8:1181–1185.
50. Discern Online, Discern quality criteria for consumer health information. www.discern.org.uk/; accessed November 2004.

51. Health Summit Working Group, Criteria for assessing the quality of health information on the Internet. www.bmlweb.org/internet_medical_critere.pdf; accessed November 2004.
52. Health on the Net Foundation, HON code of conduct (HONcode) for medical and health Web sites. <http://www.hon.ch/HONcode/>; accessed November 2004.
53. Harris J. Report on national assessment of consumer health information—demand and delivery. Reference Point Foundation, <http://odphp.osophs.dhhs.gov/harris.htm>; accessed November 2004.
54. Davis TC, Crouch MA, Wills G, Miller S, Abdehou DM. The gap between patient reading comprehension and the readability of patient education materials. *J Fam Pract* 1990;31:533–538.
55. Doak CC, Doak LG, Root IH. Teaching patients with low literacy skills. Philadelphia: JB Lippincott; 1985.
56. Holt GA, Hollon JD, Hughes SE, Coyle R. OTC labels: can consumers read and understand them? *Am Pharm* 1990;NS30(11):51–54.
57. Davis TC, Mayeaux EJ, Fredrickson D, Bocchini JA Jr, Jackson RH, Murphy PW. Reading ability of parents compared with reading level of pediatric patient education materials. *Pediatrics* 1994;93:460–468.
58. Petterson T. How readable are the hospital information leaflets available to elderly patients? *Age Ageing* 1994;23:14–16.
59. Morgan PP. Illiteracy can have major impact on patients' understanding of health care information. *Can Med Assoc J* 1993;148:1196–1197.
60. Feldman SR, Quinlivan A, Williford P, Bahnson JL, Fleischer AB, Jr. Illiteracy and the readability of patient education materials. A look at Health Watch. *N C Med J* 1994;55:290–292.
61. Jubelirer SJ, Linton JC, Magnetti SM. Reading versus comprehension: implications for patient education and consent in an outpatient oncology clinic. *J Cancer Educ* 1994;9:26–29.
62. Reid JC, Klachko DM, Kardash CA, Robinson RD, Scholes R, Howard D. Why people don't learn from diabetes literature: influence of text and reader characteristics. *Patient Educ Couns* 1995;25:31–38.
63. Berner ES, Webster GD, Shugerman AA, et al. Performance of four computer-based diagnostic systems. *N Engl J Med* 1994;330:1792–1796.

Index

A

- ABEL system, causal reasoning, 75
- Adverse drug events (ADEs), hospital-based decision support, 171–173
- AIDS, trial registries, 146
- AI/Rheum system, 39
- Alerting systems, 163–166
 - for adverse drug events (ADEs), 171–173
 - Vanderbilt WizOrder CPOE, 239–241
- Antibiotic assistant, 174–175
- Artificial intelligence, 38–39
- Associations, and ontologies, 81

B

- Bayesian systems
 - Bayes' Rule, 31–33, 108
 - development of, 108–109
 - problems of, 74, 76
- Belief networks, 76
- Biological computing, 60–61
- Boolean logic, 26–29, 173–174

C

- Caduceus system, 75
- CARE computer language, 193
- Case-based reasoning, and system design, 78
- CASNET system, 64, 112–113
 - causal reasoning, 74–75
- Causal reasoning
 - and diagnostic thinking, 106
 - and system design, 74–75

- CD-ROM, patient health information on, 254–255
- Certainty factors, 34, 73
- CHF Advisor, causal reasoning, 74–75
- Clinical decision support systems (CDSS)
 - clinical trials of, 140–151
 - components of, 5, 35
 - data mining, 44–61
 - design of. *See* Design of CDSS systems
 - effectiveness of, 8–11
 - ethical and legal issues, 126–138
 - future view, 12–13
 - genetic algorithms, 8, 41, 59–60
 - hospital-based systems, 159–185
 - implementation issues, 11–12
 - input of information, 35–37
 - knowledge-based systems, 4–6, 34–40
 - and mathematical theory, 23–34
 - neural networks, 6–8, 40–41, 52–53
 - output of information, 39–40
 - compared to physician reasoning, 34–35
 - and Regenstrief Medical Record System (RMRS), 190–211
 - rule-based versus data mining, 45, 47
 - selection of system, criteria for, 13–18
 - usage, limited, 64–65

- Clinical terminology
 - capturing for system, 85–86
 - Desiderata for, 86–87
 - local, 88–89
 - vocabulary utilization problem, 86
 - Clinical trials of CDSS, 140–151
 - Columbia Registry of Medical Management Trials, 146–147
 - diversity, pros/cons, 148–149
 - importance of, 140–142
 - information intervention categories, 148
 - information interventions, effective, 149–150
 - of patient-education interventions, 148, 150–151
 - quality improvement aspects, 142–143
 - randomized controlled trials, 144–146
 - user satisfaction measures, 143–144
 - vote-counting, 148
 - Cluster analysis
 - and data mining, 57–58
 - hierarchical, example of, 58–59
 - limitations of, 57–58
 - CMIT manual, 109
 - Columbia Registry of Medical Management Trials, 146–147
 - knowledge engineering of, 147
 - Common sense reasoning, and system design, 71, 77–78
 - Computer-based Physician Order Entry (CPOE)
 - knowledge repository infrastructure, 183–185
 - Regenrief Medical Record System (RMRS), 195–199, 208–209
 - Vanderbilt University system. *See* Vanderbilt WizOrder CPOE
 - Conditional independence, 74
 - Conditional probability, 30–31, 37
 - Consumer health information systems. *See* Patient-related decision support
 - Critical Laboratory Alerting System, 162
 - Critiquing systems, 166–170
 - Current Procedural Terminology (CPT), codes, capturing, 85
- D**
- Database management systems (DBMS), 68–69
 - Data collection, HELP System methods, 176–177
 - Data entry, design issues, 88–89
 - Data mining, 44–61
 - biological computing, 60–61
 - CDSS use of, 45–47
 - compared to rule-based systems, 45, 47
 - defined, 44
 - directed data mining. *See* Supervised learning (directed data mining)
 - genetic algorithms, 59–60
 - goals of, 44
 - Kolmogorov-Smirnov test, 56–57
 - receiver operating characteristic (ROC) graph, 54–56
 - statistical pattern recognition, 45
 - systems, examples, 46
 - undirected learning. *See* Unsupervised learning
 - Decision-theoretic reasoning, and system design, 76–77
 - Decision trees, 50–51, 68
 - DENDRAL system, 65
 - Desiderata for Controlled Medical Vocabularies, 86–87
 - Design of CDSS systems, 64–92
 - cased-based reasoning, 78
 - causal reasoning, 74–75
 - clinical content and utility issues, 84–85
 - common sense reasoning, 71, 77–78
 - data entry issues, 86, 88–89
 - decision-theoretic reasoning, 76–77
 - default/common sense knowledge, 71
 - human-computer interaction issues, 90–92
 - ignorance, dealing with, 77
 - knowledge engineering process, 79
 - logic-based knowledge representation, 66–67
 - medical knowledge, application of structure to, 65–66
 - ontologies, 79–84
 - possibilistic reasoning, 77
 - probabilistic reasoning, 76

procedural knowledge
 representation, 67
 reasoning and systems, 71–72
 rule-based systems, 72–74
 and special data types, 69–71
 storage/retrieval problems, 89
 structural data representation, 68–69
 vocabulary utilization problem, 86

Deterministic reasoning, and diagnostic thinking, 106–107

Diagnosis
 defined, 99–100
 and human reasoning, 104–107
 steps in, 100–101

Diagnostic decision support systems (DDSS), 99–121
 versus CDSS, 170–171
 clinical domain selection, 112–113
 clinical user, role of, 102–103
 defined, 101
 development of, 111–112
 diagnostic algorithms, 114
 evaluations, 115–118
 future of, 120–121
 interface/vocabulary issues, 118–119
 knowledge-base construction, 113–114
 legal/ethical issues, 119–120, 126–138
 types of systems, 107–111
 user interfaces, 114–115

Diagnostic reasoning, models of, 104–107

Directed data mining. *See* Supervised learning (directed data mining)

Diverse interventions, clinical trial effects, 148–149

Domain expert, 79

Drug administration
 adverse drug event (ADEs) decision support, 171–173
 and CDSS, 6
 cost savings, 197, 200–201
 insurance plan-dependent formulary checking, 202–203
 Regenstrief Medical Record System (RMRS), 201–203

DrugInfoNet, 253

DXplain, 110

E

Electronic medical records (EMRs), 5–6, 159

Empowerment, and patient-related decision support, 250

Ethical and legal issues, 119–120, 126–138
 on appropriate use and users, 130–131
 FDA regulation, 119–120, 135–137
 future view, 137–138
 liability, 134–135
 shared decision-making, 132–133
 standards of care, 128–130
 Standard View, 127

Evaluations of systems, 115–118
 boundary/limitation evaluation, 117
 evaluation design, 115–116
 formative evaluations, 115
 lack of system effect issue, 117–118
 successful system criteria in, 116–117

EXPERT system shell, 109

F

Family Medical Guide (AMA), 254

Food and Drug Administration (FDA)
 anti-regulation arguments, 137
 and CDSS regulation, 18, 119–120, 135–137
 on software as medical device, 135–137

Formative evaluations, 115

Forms, text displays/templates, 204–206

Foundation for Informed Medical Decision Making (FIMDM), 254

Frame of reference problem, 83

G

GALEN
 as ontology-based system, 81–82
 terminology server, 88

G-CARE, Regenstrief Medical Record System (RMRS), 197–199

Gene expression data analysis, 58–59

Genetic algorithms
 and data mining, 59–60
 operation and use of, 8, 41, 59–60
 pros/cons of, 60

Germwatcher, 29

Greek Oracle model, 102, 110

H

Health on the Net (HON) Foundation, 256

HELP System, 160–180

 alerting subsystem, 163–166

 clinical data, types used, 162

 components of, 161

 critiquing subsystem, 166–168

 data collection assistance, 176–177

 diagnostic decision support

 applications, 170–175

 subsystem modules, 162

HELP2 System, 179–181

 Computer-based Physician Order Entry (CPOE), 181–185

 modules of, 179–180

HEME program, 109

Home Medical Advisor Pro™, 255

Hooper decision, 12–13

Horizon Expert Orders. *See* Vanderbilt WizOrder CPOE

Human-computer interaction

 design issues, 90–92

Hypothetical-deductive reasoning, 105

I

If-then rules, 45, 47

Ignorance, and system design, 77

Iliad, 110, 255

Inference engine, 5, 35

 programming of, 37–39

 Regenstrief Medical Record System (RMRS), 207–208

Informal logic systems of CDSS, 34

Intermountain Healthcare (IHC)

 decision support, 159–185

 for adverse drug events (ADEs), 171–173

 alerting systems, 163–166, 171–173

 for antibiotic administration, 174–175

 assisted data collection techniques, 176–177

 categories of applications, 163

 clinical programs initiative, 181–182

 and Computer-based Physician Order Entry (CPOE), 181–185

 critiquing systems, 166–170

 HELP System, 160–180

 knowledge management tools, 182–183

 knowledge repository, 182–183

 modules of, 179–180

 for nosocomial infections, 173–174

 quality of medical reports,

 assessment of, 177–179

 suggestion systems, 168–170

International Classification of Diseases (ICD), codes, capturing, 84–85

Internet

 diagnostic decision support,

 253–254

 drug information, 252–253

 general health references, 252–253

 information seeking, scope of, 252

Intuitive reasoning, informal logic systems of CDSS, 34

J

Jones criteria, sets, 23–26

K

Knowledge Authoring Tool (KAT), 183–184

Knowledge base

 construction for DDSS, 113–114

 data representation, forms of, 68–69

 default/common sense knowledge, 71, 77–78

 features of, 5, 39

 knowledge engineering, 79

 knowledge repository, 182–183

 logic-based formats, 66–67

 ontologies, 79–84

 procedural-based formats, 67

 special data types, 69–71

 temporal knowledge, 70–71

Knowledge-based systems. *See*

 Knowledge base; Rule-based systems

Knowledge Review Online (KRO), 183

Kolmogorov-Smirnov test, 56–57

L

Lack of system effect issue, 117–118

Leeds system, 64, 74, 103

Legal issues. *See* Ethical and legal issues
 Liability issue, 134–135
 Local terminology, 88–89
 Locality, 72–73
 Logical Observation Identifiers Names and Codes (LOINC), codes, capturing, 85
 Logical operations, Boolean logic, 26–29
 Logic-based knowledge formats, 66–67
 Logistic regression, 51–52, 174, 196

M

Mathematical theory and CDSS, 23–34
 Bayes' Rule, 31–33
 Boolean logic, 26–29
 versus informal logic, 34
 probability, 29–31
 set theory, 23–26
 Mayo Clinic, Internet, health information, 253–254
 McKesson Horizon Clinical system, 6, 216
 Medical Gopher, 195–199
 Medical HouseCall, 255
 Medical reports, quality, assessment of, 177–179
 Medicare, medical necessity, Medicare standards support, 203–204
 Medication orders. *See* Drug administration
 MEDITEL system, 110
 Medline, Boolean logic, 26–30
 MedlinePlus, 253
 ModifierConcept, GALEN system, 82
 MYCIN system, 29, 34, 47, 64, 70, 110, 113–114
 certainty factors, 34, 73–74

N

Nearest neighbor (NN) classifier, 53–54
 Negligence, legal issues, 134–135
 Neural networks
 and data mining, 52–53
 features of, 6–8, 40–41, 52–53
 pros/cons of, 7–8, 40–41
 Nosocomial infections, hospital-based decision support, 173–174

O

Object-oriented database management systems (OODBMS), 69
 Ontologies, 79–84
 defined, 79
 functions of, 80
 upper ontology, 80–81
 Ontology-based systems
 frame of reference problem, 83
 GALEN, 81–82
 SNOMED CT, 82, 83
 standards, lack of, 83–84
 UMLS Semantic Network, 81
 Order entry
 electronic ordering process, 216–217
 manual ordering process, 216–217
 See also Computer-based Physician Order Entry (CPOE)
 Oxford Perinatal Database, 146, 147

P

Pathfinder system, 65, 76
 Patient education, clinical trial of effects, 150–151
 Patient-related decision support, 249–258
 CD-ROMs for, 254–255
 diagnostic decision support, 253–255
 drug information, 252–253
 future view, 258
 general health references, 252–253
 health information system design, 251–252
 patient access to, 256–257
 and patient preferences, 250–251
 and patient self-efficacy, 250
 quality of information, patient evaluation of, 255–256
 Pattern recognition, statistical pattern recognition, 45
 Pediatric HouseCall, 255
 PIP (Present Illness Program), 110
 Possibilistic reasoning, and system design, 77
 Postcoordination, 89
 Precoordination, 89
 Probabilistic reasoning
 and diagnostic thinking, 105–106, 108
 and system design, 76

- Probability theory
 basic concepts, 29–31
 Bayes' Rule, 31–33
 conditional probability, 30–31, 37
 a priori probability, 31, 48–49
- Problem Knowledge Couplers (PKCs), 90–91
- Procedural-based knowledge formats, 67
- Protégé, 82–83
- Q**
- QMR, 102, 110, 119–120
- Quality improvement, and evaluation of systems, 142–143
- Quality Screening and Management (QSM), 44–45
- R**
- Randomized controlled trials
 of CDSS, 144–146
 limitations of, 145
 units of randomization, 146
- Reasoning
 case-based, 78
 causal, 74–75, 106
 certainty factors, 34, 73
 common sense, 71, 77–78
 decision-theoretic, 76–77
 deterministic, 106–107
 diagnostic, 104–107
 hypothetical-deductive, 105
 possibilistic, 77
 probabilistic, 76, 105–106, 108
- Receiver operating characteristic (ROC) graph, 54–56
- Regenstrief Medical Record System (RMRS), 190–211
 Computer-based Physician Order Entry (CPOE), 195–199, 208–209
 drug-administration support, 201–203
 inference engine, 207–208
 information formats for user, 208–209
 insurance plan-dependent formulary checking, 202–203
 knowledge base of, 206–207
 medical necessity, Medicare standards support, 203–204
 positive impacts, 200–201, 210–211
 reminders, development of, 190–199
 studies of, 192–199
 text displays/templates, 204–206
- Reminders, Regenstrief Medical Record System (RMRS), 190–199
- RESUME system, 70
- Retrieval of data, design issues, 89
- Rule-based systems
 compared to data mining, 45, 47
 design of, 72–74
- Rule chaining, 28–29
- RxList, 253
- RxMed, 253
- S**
- Self-efficacy, and patient-related decision support, 250
- Set theory
 basic concepts, 23–26
 set covering, 25–26
- Shannon Information Content, 178
- SNOMED CT, as ontology-based system, 82, 83
- Software, as medical device, FDA on, 135–137
- Standards of care, ethical aspects, 128–130
- Standard View, 127
- Statistical pattern recognition
 CDSS systems using, 46
 and data mining, 45
- Storage of data, design issues, 89
- Structured query language (SQL), 69
- Suggestion systems, 168–170
- Supervised learning (directed data mining), 47–54
 decision trees, 50–51
 logistic regression, 51–52
 nearest neighbor (NN) classifier, 53–54
 neural networks, 52–53
 training process, 47–48
- Syllogism, 27–28
- T**
- Templates, text displays/templates, 204–206

Temporal knowledge, design issues, 70–71
 Terminologies. *See* Clinical terminology
 TOPAZ system, 70
 Training, CDSS users, 16–17
 Truth-maintenance system, 75

U

UMLS Semantic Network, as ontology-based system, 81
 Unsupervised learning, 49, 57–59
 cluster analysis, 57–58
 features of, 49
 gene expression data analysis, 58–59
 Users
 appropriate, ethical aspects, 130–131
 human reasoning/system design. *See* Reasoning
 satisfaction, measurement of, 143–144
 training, 16–17
 user interfaces, 114–115

V

Vanderbilt WizOrder CPOE, 216–243
 active orders, display of, 220–221
 decision support approaches, 237–242

 education-related links, 238
 hospital ward map, 229
 implementation, critical points of, 228–237
 individual order construction, 234–235
 individual order selection, 232–234
 initiation of session, 228
 modification of orders, 221
 order creation, 218–220
 order entry interface, 217–218
 order/session completion, 235–237
 patient selection from ward census, 228–229
 pop-up alerts, 239–241
 positive impacts, 225–228
 session events, 229–232
 underlying philosophy, 222–225
 verification of orders, 221–222
 Vote-counting, clinical efficacy, 148

W

WebMD, 252–253
 WizOrder system. *See* Vanderbilt WizOrder CPOE

X

XML documents, 182–183